PB89-159065

Validation Testing of New Mechanisms with Outdoor Chamber Data. Volume 4 Appendixes to Photochemical Reaction Photolysis Rates in the UNC Outdoor Chamber

North Carolina Univ. at Chapel Hill

Prepared for

Environmental Protection Agency
Research Triangle Park, NC

Jan 89

U.S. Department of Commerce
National Technical Information Service
NTIS

TECHNICAL REPORT DATA
(Please read Instructions on the reverse before completing)

1. REPORT NO.	2.	3. RECIPIENT'S ACCESSION NO.
EPA/600/3-89/010d		PB89 159065/AS

4. TITLE AND SUBTITLE
VALIDATION TESTING OF NEW MECHANISMS WITH OUTDOOR CHAMBER DATA, VOLUME 4: APPENDIXES TO PHOTOCHEMICAL REACTION PHOTOLYSIS RATES IN THE UNC OUTDOOR CHAMBER

5. REPORT DATE
January 1989

6. PERFORMING ORGANIZATION CODE

7. AUTHOR(S)
H. E. Jeffries, K. G. Sexton, J. R. Arnold, T. L. Kale

8. PERFORMING ORGANIZATION REPORT NO.

9. PERFORMING ORGANIZATION NAME AND ADDRESS
School of Public Health
University of North Carolina
Chapel Hill, North Carolina 27514

10. PROGRAM ELEMENT NO.
A101/B/63/20/6031 (FY-88)

11. CONTRACT/GRANT NO.
Cooperative Agreement 813107

12. SPONSORING AGENCY NAME AND ADDRESS
Atmospheric Sciences Research Laboratory-RTP, NC
Office of Research and Development
U.S. Environmental Protection Agency
Research Triangle Park, North Carolina 27711

13. TYPE OF REPORT AND PERIOD COVERED
Final (4/86 - 9/88)

14. SPONSORING AGENCY CODE
EPA 600/09

15. SUPPLEMENTARY NOTES

16. ABSTRACT

This volume contains appendixes related to the computation of photolytic rates for the UNC chamber. The first appendix is the data base of climate and solar radiation variables needed by the models, including time of day, air pressure, total column ozone, total column water vapor, relative humidity, aerosol type, aerosol optical density at 500 nm, and the spectral albedo of the aerosol. The second appendix gives plots of broad band model predictions by the models compared with observed TSR and UV irradiance data for each day in the UNC data base, and includes selected examples of plots of the predicted NO_2 and HCHO-to-radicals photolysis rates inside the chamber. The third appendix lists the molecular absorption cross sections and reaction quantum yields used in the photolysis rate calculations. The final appendix includes listings of the computer code for all models and sub-models. These models run on MS/PC-DOS type machines and are available on diskettes from the authors for distribution costs. The in-chamber photolysis rates for twenty species, computed every 16 minutes for each day in the UNC data base are also available on diskettes for distribution costs.

17. KEY WORDS AND DOCUMENT ANALYSIS

a. DESCRIPTORS	b. IDENTIFIERS/OPEN ENDED TERMS	c. COSATI Field/Group

18. DISTRIBUTION STATEMENT	19. SECURITY CLASS (This Report)	21. NO. OF PAGES
RELEASE TO PUBLIC	UNCLASSIFIED	142
	20. SECURITY CLASS (This page) UNCLASSIFIED	22. PRICE PC$21.95/MF$6.95

EPA Form 2220-1 (9-73)

PB89-159065

EPA/600/3-89/010d
January 1989

Validation Testing of New Mechanisms With Outdoor Chamber Data
Volume 4: Appendixes To Photochemical Reaction Photolysis Rates in the UNC Outdoor Chamber

H. E. Jeffries, K. G. Sexton,
J. R. Arnold, T. L. Kale

Department of Environmental Sciences
and Engineering
School of Public Health
University of North Carolina
Chapel Hill, N.C. 27514

In partial fullfilment of
Cooperative Agreement 813107

Project Officer
Marcia Dodge

Atmospheric Chemistry and Physics Division
Atmospheric Sciences Research Laboratory
Research Triangle Park, N.C. 27711

ATMOSPHERIC RESEARCH AND EXPOSURE ASSESSMENT LABORATORY
OFFICE OF RESEARCH AND DEVELOPMENT
U.S. ENVIRONMENTAL PROTECTION AGENCY
RESEARCH TRIANGLE PARK, NC 27711

REPRODUCED BY
U.S. DEPARTMENT OF COMMERCE
NATIONAL TECHNICAL INFORMATION SERVICE
SPRINGFIELD, VA. 22161

DISCLAIMER

This report has been reviewed by the U. S. Environmental Protection Agency, and approved for publication. Approval does not signify that the contents necessarily reflect the views and policies of the U. S. Environmental Protection Agency, nor does mention of trade names or commercial products constitute endorsement or recommendation for use.

Abstract

This volume contains appendixes related to the computation of in-chamber photolytic rates for the UNC chamber.

The first appendix is the data base of climate and solar radiation variables needed by the models, including time of day, air pressure, total column ozone, total column water vapor, relative humidity, aerosol type, aerosol optical density at 500 nm, and the spectral albedo of the aerosol.

The second appendix gives plots of broad band model predictions by the models compared with observed site TSR and UV irradiance data for each day in the UNC data base, and includes selected examples of plots of the predicted in-chamber NO_2 and HCHO to radicals photolysis rates inside the chamber.

The third appendix lists the molecular absorption cross sections and reaction quantum yields used in the photolysis rate calculations.

The final appendix includes listings of the computer code for all models and sub-models used. Included in these listings are:

- a highly accurate solar coordinate model (SOLCALC),
- an aerosol optical model (AEROMOD),
- two broad band models for predicting the TSR and UV radiometer irradiances (BBMODEL),
- a spectral spherical irradiance model for predicting the actinic flux components (JSMODEL),
- a TFE film transmission model,
- a total chamber transmission model, a reflective floor model, and a treatment for clouds (CHMMODEL).

The main model code is given in CHMJSMOD.

The final program listed is CHMRATE, which uses actinic flux spectra from the CHMMODEL and the absorption cross-sections and quamtum yields in the third appendix to compute the in-chamber rates.

These models run on MS/PC-DOS type machines and are available on disketts from the authors for distribution costs.

The in-chamber photolysis rates for twenty species, computed every 16 minutes for each day in the SegFile data base are also available on disketts for distribution costs.

Contents

A SolClim Data Base 1

B Model Plots 14

Sample Clear Model Plots 15

Cloudy Model Plots 18

Sample In-Chamber NO_2 and HCHO Rates Plots 52

C CsQy Files 55

Nitrogen Dioxide 55

Nitrogen Dioxide, cont 56

Nitrous Acid 57

Nitrate Radical 58

Nitrate Radical to NO_2 59

Ozone, UV 60

Ozone, visible 61

Hydrogen Peroxide 62

Hydro-Methyl Peroxide 62

Formaldehyde to Radicals 63

Formaldehyde to Stable Products 64

Acetaldehyde to Radicals 65

Propyaldehyde to Radicals 65

Benzaldehyde 66

Acetone 67

Methyl-Ethyl-Ketone 67

Glyoxal, below 350 NM 68

Glyoxal, above 350 NM 69

Methy Glyoxyl, below 350 NM 71

Methyl Glyoxyl, above 350 NM 72

CAL Aromatic Unknown 73

D Flux Model Code 74

SOLCALC 74

AEROMOD 81

BBMODEL 87

JSMODEL 92

CHMMODEL 98

CHMJSMOD 105

E Chamber Rate Code 111

Spectral Integration 111

CHMRATE 116

Acknowledgements

We very much appreciate the extra efforts of Dr. Marcia Dodge that allow us to acquire the Licor automatic scanning spectroradiometer used in this project. This instrument allowed us to make measurements that confirmed the accuracy of the model described here.

The assistance of Allen Hendler, graduate student at UNC, in clarifying the spectral transmission of clouds in the UV region is greatly appreciated.

We also thank Ken Daisy, student at UNC, for his careful efforts in performing quality assurance activities on all of our data files.

A

SolClim Data Base

The SOLCLIM data base is used by the broadband and spectral irradiance models as the source of meteorological conditions for each day in the UNC SegFile data base. See the text in Volume 3 for a discussion of the use of these data. The data fields are described below.

`Tcorr`	minutes to add to data clock to produce EDT
`Press`	atmospheric pressure at site
`Col03`	total atmospheric column ozone in cm
`ColWat`	total atmospheric column water vapor in cm
`Amc`	aerosol model, 0=user specified parameters, 1=Standard Rural parameters, 2=Standard Urban parameters (see AEROMOD) listing in Appendix D below for a descriptextitemon of parameters).
`odau`	upper layer aerosol optical density
`odal`	lower layer aerosol optical density
`omega`	lower layer aerosol single scattering albedo
`RH`	site ambient relative humidity at given times
`EndTn`	ending time of five intervals for cloud transmissions
`tsrT`	the ratio between the "clear" model and the site TSR radiometer, averaged over the time interval beginning at `EndTn-1` and ending at `EndTn`
`uvT`	the ratio between the "clear" model and the site UV radiometer, averaged over the time interval beginning at `EndTn-1` and ending at `EndTn`

The first nine items are used to compute the "Clear" sky irradiances and the last two items are used to correct the "clear" model for observed cloud effects.

	A	B	C	K	L	M	N	O	P	Q	R	T	U	V	W	X	Y	Z	AA	AB	AC	AD	AE	AF	AG	AH	AI	AJ	AK	AL
3	3	6	5	5	7	6	6	4	5	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6
4				Tcorr	Press	ColO3	colWat	Amc	odau	odal	omega	RH	RH	RH	RH	EndT1	tsrT	uvT	EndT2	tsrT	uvT	EndT3	tsrT	uvT	EndT4	tsrT	uvT	EndT5	tsrT	uvT
5	Dir	Filename		mins	mbars	cm	cm					0700	1000	1300	1600		to ENDT1			to ENDT2			to ENDT3			to ENDT4			to ENDT5	
6	77\	MY1877	.SEG	0	1000.0	0.339	3.50	2	0.025	0.304	0.880	0.81	0.49	0.38	0.32	9.0	0.847	0.847	12.0	0.821	0.821	14.5	0.833	0.833	17.0	0.784	0.784	19.0	0.772	0.772
7	77\	JN0377	.SEG	0	1000.8	0.334	2.90	0	0.025	0.330	0.600	0.81	0.58	0.39	0.33	9.0	1.066	1.066	11.2	0.920	0.869	12.0	0.920	0.860	18.0	0.902	0.902	19.0	1.000	1.000
8	77\	JL1877	.SEG	0	1004.2	0.309	3.00	0	0.025	0.454	0.600	0.87	0.65	0.49	0.42	10.0	1.087	1.087	12.0	1.005	1.005	14.5	0.944	0.944	15.5	0.530	0.530	19.0	0.893	0.893
9	77\	JL2077	.SEG	0	1001.4	0.309	3.00	0	0.025	0.457	0.600	0.85	0.58	0.39	0.37	10.5	1.091	1.091	12.8	0.969	0.969	14.0	0.799	0.799	16.5	0.727	0.727	19.0	0.725	0.725
10	77\	JL2777	.SEG	0	1004.2	0.306	3.10	1	0.025	0.469	0.946	0.78	0.52	0.43	0.40	9.0	1.070	1.040	12.0	1.010	0.970	13.5	0.786	0.774	16.0	0.952	0.837	19.0	0.724	0.681
11	77\	JL2877	.SEG	0	1006.5	0.306	2.60	1	0.025	0.470	0.946	0.90	0.42	0.28	0.29	10.4	1.075	1.075	12.8	1.007	1.007	14.5	0.941	0.941	16.0	0.812	0.812	19.0	0.928	0.928
12	77\	AU0877	.SEG	0	1001.8	0.302	3.00	0	0.025	0.488	0.600	0.90	0.72	0.54	0.57	10.0	1.110	1.110	12.5	1.008	1.008	14.5	0.633	0.633	16.0	0.401	0.401	19.0	0.853	0.853
13	77\	ST1477	.SEG	0	1000.8	0.281	2.53	1	0.025	0.307	0.946	0.97	0.79	0.63	0.55	9.0	0.841	0.841	11.0	0.503	0.503	13.0	0.686	0.686	15.0	0.712	0.712	19.0	0.699	0.699
14	77\	ST1977	.SEG	0	995.4	0.279	2.44	0	0.025	0.284	0.700	0.98	0.82	0.65	0.79	9.0	0.939	0.939	11.5	0.952	0.952	13.0	0.769	0.769	14.0	0.640	0.640	19.0	0.751	0.751
15	77\	OC1277	.SEG	0	1000.1	0.271	2.07	1	0.025	0.194	0.946	0.98	0.80	0.86	0.96	9.0	0.207	0.627	11.5	0.360	0.504	14.5	0.221	0.322	18.0	0.180	0.289	19.0	0.000	2.042
16	77\	OC1877	.SEG	0	997.4	0.271	1.99	1	0.025	0.177	0.946	0.98	0.51	0.37	0.39	9.0	1.083	1.083	12.0	0.950	0.950	14.0	0.801	0.801	15.0	0.911	0.911	19.0	0.736	0.736
17	77\	OC2277	.SEG	0	1007.9	0.272	1.94	1	0.025	0.172	0.946	0.98	0.65	0.41	0.36	9.0	1.000	1.000	13.5	0.948	0.948	15.0	0.433	0.433	18.0	0.040	0.040	19.0	1.000	1.000
18	77\	OC2477	.SEG	0	1014.0	0.273	1.92	1	0.025	0.169	0.946	0.96	0.93	0.70	0.81	9.0	0.478	0.478	12.0	0.824	0.824	15.0	0.766	0.766	18.0	1.000	1.000	19.0	1.000	1.000
19	77\	NO0977	.SEG	0	1000.8	0.278	1.72	1	0.025	0.148	0.946	0.93	0.73	0.55	0.57	9.0	1.214	1.214	12.0	0.939	0.939	15.0	0.873	0.873	18.0	0.740	0.740	19.0	1.000	1.000
20	77\	NO1277	.SEG	0	1006.2	0.279	1.72	1	0.025	0.144	0.946	0.81	0.31	0.23	0.27	9.0	1.000	1.000	12.0	1.087	1.087	14.0	1.060	1.060	15.0	0.573	0.573	19.0	0.584	0.584
21	77\	NO1977	.SEG	0	1010.2	0.281	1.72	1	0.025	0.135	0.946	0.59	0.35	0.24	0.22	9.0	1.000	1.000	12.0	1.080	1.080	14.0	0.821	0.821	15.0	0.803	0.803	19.0	0.473	0.473
22	77\	NO2077	.SEG	0	1014.0	0.282	1.72	1	0.025	0.133	0.946	0.82	0.54	0.35	0.33	9.0	1.050	1.050	12.0	1.006	1.000	15.0	0.797	0.797	18.0	0.598	0.598	19.0	1.000	1.000
23	77\	DE2677	.SEG	0	998.1	0.293	1.72	1	0.025	0.107	0.946	0.53	0.43	0.38	0.36	9.0	1.000	1.000	12.0	1.060	1.060	15.0	0.925	0.925	18.0	0.802	0.802	19.0	1.000	1.000
24	78\	JA1078	.SEG	5	1005.8	0.298	1.10	1	0.025	0.117	0.946	0.56	0.45	0.33	0.30	9.0	1.000	1.000	15.0	1.020	1.020	17.0	0.899	0.899	18.0	1.000	1.000	19.0	1.000	1.000
25	78\	JA2878	.SEG	5	999.1	0.308	1.00	1	0.025	0.200	0.946	0.63	0.59	0.39	0.44	9.0	1.000	1.000	12.0	1.000	1.000	15.0	0.968	0.968	18.0	0.810	1.050	19.0	1.000	1.000
26	78\	JA2978	.SEG	20	1003.5	0.309	1.00	1	0.025	0.200	0.946	0.77	0.58	0.38	0.36	9.0	1.000	1.136	12.0	1.000	1.094	15.0	1.040	1.040	18.0	1.104	1.050	19.0	1.000	1.000
27	78\	FE2778	.SEG	20	1004.0	0.325	1.07	1	0.025	0.200	0.946	0.84	0.52	0.38	0.30	11.0	0.976	0.976	14.0	0.887	0.887	15.0	0.942	0.942	18.0	0.995	0.995	19.0	1.500	1.500
28	78\	MR0678	.SEG	20	1005.2	0.327	1.10	2	0.025	0.168	0.880	0.69	0.60	0.25	0.27	9.0	0.353	0.353	11.0	0.787	0.787	12.0	0.716	0.716	15.0	0.863	0.863	19.0	0.889	0.889
29	78\	MR3178	.SEG	20	1003.5	0.335	1.44	2	0.025	0.215	0.880	0.98	0.45	0.30	0.27	9.5	0.894	0.894	12.0	0.805	0.805	14.0	0.847	0.847	15.0	0.840	0.840	19.0	0.840	0.840
30	78\	JN1678	.SEG	20	1010.6	0.329	3.02	0	0.025	0.353	0.600	0.78	0.52	0.47	0.43	10.0	0.828	0.828	12.5	0.993	0.993	15.0	0.962	0.962	18.0	0.993	0.993	19.0	1.000	1.000
31	78\	JN1878	.SEG	20	1006.9	0.328	3.50	0	0.025	0.360	0.600	0.84	0.67	0.55	0.53	10.5	0.860	0.907	12.3	0.873	0.868	13.8	0.913	0.918	15.0	0.851	0.883	19.0	0.950	0.950
32	78\	JN2178	.SEG	20	1001.4	0.326	3.50	0	0.025	0.370	0.600	0.87	0.67	0.52	0.77	8.5	0.714	0.714	11.1	0.929	0.929	13.3	0.948	0.948	15.0	0.693	0.693	19.0	0.469	0.469
33	78\	JL0178	.SEG	20	998.4	0.319	3.50	0	0.025	0.403	0.600	0.79	0.65	0.52	0.55	11.2	0.908	0.908	14.2	0.957	0.957	15.8	0.924	0.924	16.8	0.849	0.849	19.0	1.000	1.000
34	78\	JL2178	.SEG	25	1005.2	0.308	3.50	0	0.025	0.459	0.700	0.87	0.53	0.48	0.44	10.0	1.016	1.054	13.0	1.049	1.013	15.5	1.039	1.008	18.0	1.028	0.963	19.0	1.000	1.000
35	78\	JL2278	.SEG	25	1004.5	0.308	3.50	0	0.025	0.461	0.600	0.91	0.65	0.54	0.54	9.0	0.941	1.105	12.0	1.049	1.003	15.0	0.989	0.972	18.0	0.996	0.947	19.0	1.000	1.000
36	78\	JL2478	.SEG	20	1004.2	0.307	3.50	0	0.025	0.464	0.500	0.85	0.72	0.49	0.63	9.0	0.820	1.060	12.0	0.935	0.920	14.5	0.963	0.958	18.0	0.883	0.848	19.0	1.000	1.000
37	78\	JL2778	.SEG	20	998.7	0.306	3.50	0	0.025	0.469	0.700	0.90	0.74	0.63	0.57	9.0	0.500	0.600	11.5	0.416	0.507	13.0	0.373	0.430	14.0	0.730	0.759	19.0	0.856	0.869
38	78\	JL3078	.SEG	20	997.7	0.305	3.50	0	0.025	0.473	0.600	0.90	0.63	0.50	0.46	10.5	0.710	0.858	13.8	1.007	0.984	15.0	0.949	0.924	18.0	0.992	0.958	19.0	1.037	1.296
39	78\	AU0578	.SEG	20	1006.5	0.303	3.50	2	0.025	0.483	0.880	0.97	0.77	0.77	0.70	9.8	0.627	0.758	12.0	0.821	0.812	13.0	0.524	0.496	15.0	0.710	0.721	19.0	0.613	0.664
40	78\	AU0678	.SEG	20	1006.9	0.303	3.50	2	0.025	0.485	0.880	0.97	0.80	0.94	0.85	9.0	0.516	0.726	12.5	0.650	0.670	15.0	0.300	0.300	18.0	0.783	0.749	19.0	1.000	1.000
41	78\	AU0878	.SEG	20	1007.2	0.302	3.50	2	0.025	0.488	0.880	0.97	0.79	0.66	0.59	10.5	0.446	0.537	12.1	0.972	0.954	15.0	0.919	0.900	18.0	0.943	0.878	19.0	1.000	1.000
42	78\	AU1178	.SEG	20	1002.5	0.301	3.50	0	0.025	0.493	0.700	0.93	0.77	0.57	0.65	9.5	0.500	0.666	11.5	0.590	0.687	15.0	0.782	0.836	18.0	0.514	0.615	19.0	1.000	1.000
43	78\	AU1478	.SEG	20	1003.8	0.300	3.50	0	0.025	0.498	0.700	0.98	0.85	0.70	0.65	9.0	0.638	0.909	11.8	0.686	0.731	15.0	0.910	0.897	18.0	0.750	0.767	19.0	1.000	1.000
44	78\	AU1578	.SEG	30	1004.2	0.300	3.50	0	0.025	0.499	0.800	0.98	0.77	0.61	0.18	9.0	0.700	0.903	12.2	1.000	0.959	13.5	0.922	0.901	15.0	0.575	0.637	19.0	0.906	0.853
45	78\	AU1678	.SEG	30	1001.8	0.300	3.48	0	0.025	0.497	0.700	0.94	0.70	0.60	0.59	9.0	0.636	0.853	12.8	0.997	0.968	14.6	0.940	0.897	18.0	0.733	0.790	19.0	1.000	1.000
46	78\	AU1778	.SEG	30	999.4	0.299	3.45	0	0.025	0.490	0.700	0.91	0.70	0.47	0.51	9.0	0.659	0.659	12.0	0.752	0.752	15.0	0.870	0.870	18.0	0.916	0.916	19.0	1.000	1.000
47	78\	AU1878	.SEG	30	1000.4	0.298	3.42	0	0.025	0.484	0.600	0.84	0.61	0.57	0.50	9.0	0.897	1.187	13.0	1.072	1.039	15.0	1.000	1.000	18.0	1.000	0.950	19.0	1.000	1.000
48	78\	AU2178	.SEG	30	1006.2	0.296	3.32	0	0.025	0.465	0.800	0.81	0.58	0.46	0.51	9.0	0.766	0.948	13.0	1.055	1.006	15.0	1.021	0.988	18.0	1.065	0.964	19.0	1.000	1.000
49	78\	AU2378	.SEG	30	1005.8	0.295	3.26	0	0.025	0.452	0.600	0.97	0.58	0.48	0.42	9.0	0.861	1.118	13.0	1.067	1.000	15.0	1.036	0.904	18.0	1.063	0.905	19.0	1.000	1.000
50	78\	AU2478	.SEG	20	1003.5	0.295	3.23	0	0.025	0.445	0.700	0.93	0.58	0.45	0.44	9.0	1.024	1.156	13.0	1.075	0.991	15.0	0.991	0.899	18.0	1.000	0.882	19.0	1.000	1.000
51	78\	AU2578	.SEG	20	1000.1	0.294	3.19	0	0.025	0.439	0.500	0.90	0.65	0.49	0.49	9.0	0.886	1.173	13.2	0.946	0.931	15.0	0.738	0.750	18.0	0.870	0.906	19.0	1.000	1.000
52	78\	ST0878	.SEG	20	999.4	0.285	2.73	2	0.025	0.347	0.880	0.84	0.63	0.54	0.52	9.0	0.682	0.801	12.2	0.476	0.521	15.0	0.842	0.745	18.0	0.898	0.758	19.0	1.000	1.000
53	78\	ST1478	.SEG	20	1006.5	0.281	2.53	0	0.025	0.307	0.700	0.93	0.76	0.62	0.60	10.0	0.832	0.963	11.0	0.565	0.705	13.0	0.705	0.797	15.0	0.906	0.953	19.0	0.876	0.964
54	78\	ST1578	.SEG	20	1003.5	0.280	2.50	2	0.025	0.300	0.850	0.90	0.81	0.79	0.74	9.0	0.190	0.353	12.0	0.293	0.373	15.0	0.298	0.383	18.0	0.431	0.535	19.0	1.000	1.000
55	78\	ST1878	.SEG	20	1004.2	0.279	3.00	0	0.025	0.288	0.700	0.98	0.74	0.56	0.52	11.5	0.882	0.972	12.5	0.654	0.704	13.5	0.789	0.821	15.0	0.800	0.803	19.0	0.819	0.834
56	78\	ST1978	.SEG	20	1003.1	0.279	3.00	0	0.025	0.284	0.600	0.97	0.70	0.47	0.49	9.0	0.733	1.065	12.0	0.902	0.978	15.0	0.913	0.946	18.0	0.890	0.918	19.0	1.000	1.000
57	78\	ST2178	.SEG	20	1003.5	0.278	2.60	0	0.025	0.276	0.700	0.98	0.69	0.63	0.59	9.0	0.344	0.668	12.0	0.406	0.528	15.0	0.725	0.738	18.0	0.783	0.658	19.0	1.000	1.000

	A	B	C	K	L	M	N	O	P	Q	R	T	U	V	W	X	Y	Z	AA	AB	AC	AD	AE	AF	AG	AH	AI	AJ	AK	AL
58	78\	OC0378	.SEG	20	1001.1	0.274	2.50	0	0.025	0.228	0.700	0.98	0.80	0.62	0.58	9.0	0.667	1.174	13.0	0.910	0.966	14.1	0.570	0.649	16.0	0.912	0.923	19.0	0.490	0.761
59	78\	OC1278	.SEG	30	1001.8	0.271	3.80	2	0.025	0.194	0.880	0.98	0.73	0.52	0.49	9.0	0.480	0.797	12.0	0.890	0.928	16.0	0.957	0.949	18.0	0.986	1.006	19.0	1.000	1.000
60	78\	OC1378	.SEG	30	1001.1	0.271	3.80	2	0.025	0.190	0.880	0.98	0.79	0.60	0.56	9.0	0.560	1.145	12.0	0.627	0.769	15.0	0.767	0.842	18.0	0.428	0.691	19.0	1.000	1.000
61	78\	OC1778	.SEG	30	1009.9	0.270	2.00	2	0.025	0.178	0.880	0.98	0.69	0.60	0.64	9.0	0.522	0.905	11.5	0.941	1.009	13.0	0.729	0.820	14.0	0.871	0.905	19.0	0.632	0.758
62	78\	OC1878	.SEG	30	1012.3	0.271	2.20	1	0.025	0.177	0.946	0.98	0.69	0.43	0.45	9.0	0.458	0.866	12.0	0.935	0.983	15.0	0.993	0.998	18.0	0.973	1.036	19.0	1.000	1.000
63	78\	OC2078	.SEG	30	1000.8	0.271	1.97	1	0.025	0.174	0.946	0.83	0.46	0.39	0.38	9.5	0.467	0.896	12.0	0.963	1.004	15.0	0.945	0.970	18.0	1.000	1.029	19.0	1.000	1.000
64	78\	OC2178	.SEG	36	1006.2	0.272	2.75	2	0.025	0.173	0.880	0.96	0.68	0.40	0.38	9.0	0.667	0.904	12.0	0.981	1.005	15.0	0.962	0.985	18.0	0.990	1.001	19.0	1.000	1.000
65	78\	OC2278	.SEG	36	1006.9	0.272	2.75	2	0.025	0.172	0.880	0.98	0.73	0.47	0.44	9.0	0.524	0.891	12.0	0.937	0.972	15.0	0.954	0.965	18.0	0.951	0.980	19.0	1.000	1.000
66	78\	OC2578	.SEG	40	1003.1	0.273	2.20	1	0.025	0.260	0.946	0.96	0.67	0.42	0.36	9.0	0.800	0.846	12.0	1.020	1.007	15.0	1.004	0.975	18.0	0.982	0.980	19.0	1.000	1.000
67	78\	OC2978	.SEG	20	1008.2	0.274	2.00	2	0.025	0.163	0.880	0.93	0.68	0.49	0.47	9.0	0.118	0.730	12.0	0.400	0.575	13.0	0.677	0.812	16.0	0.925	0.884	19.0	0.964	0.880
68	78\	NO0278	.SEG	20	1006.9	0.276	2.00	2	0.025	0.157	0.880	0.98	0.72	0.46	0.39	9.0	0.600	1.293	12.0	0.873	0.910	16.0	0.910	0.899	18.0	0.882	0.900	19.0	1.000	1.000
69	78\	NO0778	.SEG	20	999.4	0.277	1.74	1	0.025	0.151	0.946	0.98	0.73	0.51	0.55	9.0	0.643	1.318	12.0	0.858	0.981	13.8	0.906	0.975	16.0	0.583	0.762	19.0	0.842	1.009
70	79\	MY2179	.SEG	0	1000.0	0.338	2.79	2	0.025	0.309	0.880	0.97	0.87	0.79	0.90	9.0	1.000	1.000	12.0	1.000	1.000	15.0	1.000	1.000	18.0	1.000	1.000	19.0	1.000	1.000
71	79\	MY3179	.SEG	0	1000.0	0.335	2.87	2	0.025	0.325	0.880	0.98	0.90	0.65	0.72	9.0	0.418	0.635	11.0	0.398	0.560	15.0	0.579	0.764	18.0	1.000	1.000	19.0	1.000	1.000
72	79\	JN0179	.SEG	0	1002.1	0.335	2.88	0	0.025	0.327	0.700	0.98	0.85	0.69	0.69	10.0	0.725	0.998	11.0	0.790	1.018	12.5	0.679	0.928	15.0	0.666	0.880	19.0	0.792	1.022
73	79\	JN0279	.SEG	0	1000.8	0.334	2.89	0	0.025	0.280	0.700	0.98	0.69	0.63	0.57	9.0	1.000	1.000	12.0	1.000	1.000	15.0	0.900	0.900	17.8	0.900	0.879	19.0	1.000	1.106
74	79\	JN1279	.SEG	0	1001.8	0.331	3.50	1	0.025	0.345	0.946	0.90	0.55	0.50	0.35	11.7	0.898	0.989	12.2	0.781	0.870	15.0	0.920	1.009	18.0	0.831	1.004	19.0	1.000	1.000
75	79\	JN1379	.SEG	0	1006.5	0.331	2.80	2	0.025	0.347	0.880	0.90	0.45	0.42	0.40	9.0	1.157	1.130	12.0	0.962	1.003	15.0	0.890	0.996	18.0	0.783	0.982	19.0	1.000	1.000
76	79\	JN1479	.SEG	0	1010.9	0.330	2.99	2	0.025	0.348	0.880	0.87	0.62	0.52	0.50	11.2	0.930	0.970	12.0	0.688	0.733	15.0	0.818	0.917	18.0	0.801	0.834	19.0	1.000	1.000
77	79\	JN2079	.SEG	0	1010.9	0.327	4.00	1	0.025	0.367	0.946	0.84	0.48	0.44	0.44	9.5	0.717	0.842	12.0	0.885	0.966	15.0	0.888	0.987	18.0	0.847	0.954	19.0	1.000	1.000
78	79\	JN2779	.SEG	0	1009.9	0.322	3.20	1	0.025	0.450	0.946	0.87	0.62	0.58	0.52	9.0	0.932	0.923	12.2	0.997	1.007	15.0	0.767	0.862	18.0	0.889	1.000	19.0	1.000	1.000
79	79\	JN2979	.SEG	0	1001.4	0.321	3.23	2	0.025	0.397	0.880	0.87	0.62	0.55	0.58	9.0	0.857	0.991	12.0	0.825	0.821	13.1	0.746	0.789	13.8	0.705	0.724	19.0	0.200	0.200
80	79\	JL2179	.SEG	0	1004.2	0.308	3.50	2	0.025	0.459	0.880	0.98	0.94	0.85	0.72	9.0	0.333	0.546	12.0	0.352	0.496	14.2	0.650	0.820	18.0	0.394	0.553	19.0	0.182	0.374
81	79\	JL2979	.SEG	0	997.7	0.306	3.50	2	0.025	0.472	0.880	0.90	0.74	0.57	0.59	9.0	0.567	0.744	12.0	0.766	0.826	14.5	0.828	0.887	15.5	0.575	0.649	19.0	0.379	0.461
82	79\	JL3079	.SEG	0	998.7	0.305	3.50	2	0.025	0.473	0.880	0.84	0.76	0.74	0.77	10.0	0.511	0.677	12.5	0.442	0.524	15.0	0.620	0.706	18.0	0.100	0.100	19.0	0.100	0.100
83	79\	JL3179	SEG	0	1000.4	0.305	4.00	0	0.025	0.475	0.700	0.98	0.69	0.61	0.55	9.0	1.167	1.107	11.0	1.000	1.000	15.0	1.000	1.000	18.0	1.000	1.000	19.0	1.000	1.000
84	79\	AU0179	.SEG	0	999.1	0.305	4.00	0	0.025	0.477	0.600	0.97	0.72	0.61	0.59	9.0	1.021	1.100	12.8	0.953	0.966	15.0	0.047	0.051	18.0	1.000	1.000	19.0	1.000	1.000
85	79\	AU0279	.SEG	0	997.7	0.304	4.00	0	0.025	0.478	0.600	0.97	0.74	0.68	0.61	9.0	1.042	1.100	12.5	0.920	0.985	15.2	0.870	0.910	18.0	1.000	1.000	19.0	1.000	1.000
86	79\	AU0479	.SEG	0	1001.8	0.304	4.00	0	0.025	0.482	0.600	0.90	0.70	0.57	0.52	9.0	1.018	1.135	12.0	0.980	0.928	15.0	0.986	0.920	18.0	1.000	1.000	19.0	1.000	1.000
87	79\	AU0579	.SEG	0	1002.8	0.303	4.00	0	0.025	0.483	0.500	0.93	0.67	0.52	0.47	9.0	1.100	1.100	12.0	0.950	1.050	15.0	0.950	1.050	18.0	0.950	1.040	19.0	1.000	1.000
88	79\	AU1379	.SEG	0	1004.2	0.301	3.50	2	0.025	0.496	0.880	0.96	0.64	0.50	0.44	9.0	1.018	1.072	12.0	1.036	1.022	15.0	0.949	0.977	18.0	0.983	0.981	19.0	1.000	1.000
89	79\	AU1579	.SEG	0	1002.8	0.300	3.50	0	0.025	0.499	0.700	0.87	0.64	0.58	0.44	9.0	0.809	0.983	12.5	0.474	0.590	15.0	0.778	0.874	18.0	0.938	1.007	19.0	1.000	1.000
90	79\	AU1679	.SEG	0	1007.5	0.300	4.00	0	0.025	0.497	0.700	0.78	0.53	0.43	0.43	9.0	1.180	1.170	12.0	1.054	1.076	15.0	0.997	1.047	18.0	1.044	1.088	19.0	1.100	1.258
91	79\	ST0779	.SEG	0	995.0	0.285	3.90	0	0.025	0.400	0.500	0.98	0.65	0.55	0.57	9.0	1.007	1.080	12.8	0.927	0.997	14.0	0.817	0.898	18.0	0.916	0.972	19.0	1.000	1.000
92	79\	OC0979	.SEG	0	995.7	0.272	3.00	2	0.025	0.205	0.880	0.84	0.62	0.51	0.41	9.0	0.964	1.070	12.0	0.912	1.005	15.0	0.895	0.970	18.0	0.866	0.970	19.0	1.000	1.000
93	79\	OC1479	.SEG	0	1006.9	0.270	3.00	1	0.025	0.250	0.946	0.96	0.57	0.49	0.46	9.0	1.115	1.234	12.0	0.959	1.021	16.0	0.984	1.001	18.0	0.875	1.000	19.0	1.000	1.000
94	79\	OC1879	.SEG	0	1007.5	0.271	3.00	2	0.025	0.177	0.880	0.93	0.71	0.49	0.50	9.0	0.913	1.363	13.8	0.852	0.973	15.2	0.544	0.732	18.0	0.772	0.979	19.0	1.000	1.000
95	79\	OC2079	.SEG	0	1006.2	0.271	2.50	1	0.025	0.270	0.946	0.97	0.67	0.58	0.56	10.5	0.693	0.889	13.5	0.899	0.932	15.0	0.864	0.868	18.0	0.791	1.014	19.0	1.000	1.000
96	79\	OC2479	.SEG	0	995.4	0.273	2.20	1	0.025	0.270	0.946	0.86	0.57	0.46	0.45	9.0	1.095	1.070	12.0	0.970	1.026	15.0	0.975	0.988	18.0	0.905	0.975	19.0	1.000	1.000
97	79\	DE1179	.SEG	0	1001.4	0.289	1.00	1	0.025	0.200	0.946	0.98	0.98	0.98	0.98	9.0	1.050	1.160	12.0	1.120	1.161	16.0	1.127	1.135	18.0	1.000	1.140	19.0	1.000	1.000
98	80\	JN0480	.SEG	0	999.8	0.334	2.50	0	0.025	0.332	0.700	0.79	0.55	0.42	0.39	9.0	1.123	1.070	12.0	1.069	0.969	15.0	1.054	1.001	18.0	1.053	0.999	19.0	1.077	1.121
99	80\	JN1280	.SEG	0	1007.9	0.331	3.00	0	0.025	0.345	0.500	0.93	0.55	0.47	0.42	9.0	1.143	1.108	12.2	1.095	0.962	15.0	0.960	0.859	16.0	0.612	0.636	19.0	1.087	1.013
100	80\	JN1480	.SEG	0	1000.4	0.330	3.90	0	0.025	0.348	0.700	0.93	0.52	0.47	0.46	9.0	0.831	0.831	12.0	1.017	1.017	15.7	1.022	1.022	18.0	0.920	0.920	19.0	0.758	0.758
101	80\	JN1580	.SEG	0	997.0	0.330	3.00	0	0.025	0.350	0.700	0.79	0.55	0.40	0.39	9.0	0.938	0.938	12.5	0.887	0.887	15.0	0.024	0.024	18.0	1.000	1.000	19.0	1.000	1.000
102	80\	JN1780	.SEG	0	1002.8	0.329	3.03	0	0.025	0.357	0.700	0.68	0.55	0.45	0.39	9.0	0.763	0.763	12.0	0.757	0.757	15.0	0.974	0.974	18.0	1.090	1.090	19.0	1.121	1.121
103	80\	JN2080	.SEG	0	999.1	0.327	2.80	0	0.025	0.367	0.700	0.97	0.72	0.53	0.28	9.0	0.913	0.913	12.0	0.852	0.852	15.0	0.905	0.905	18.0	1.049	1.049	19.0	1.152	1.152
104	80\	JN2180	.SEG	0	1003.5	0.326	2.80	0	0.025	0.370	0.700	0.65	0.48	0.41	0.42	9.0	0.795	0.795	12.0	0.983	0.983	15.0	0.849	0.849	18.0	1.019	1.019	19.0	1.294	1.294
105	80\	JL0680	.SEG	0	1001.8	0.315	3.38	0	0.025	0.426	0.700	0.98	0.85	0.70	0.68	9.0	0.956	0.962	11.5	0.546	0.569	14.5	0.836	0.800	17.0	0.760	0.754	19.0	0.792	0.882
106	80\	JL0980	.SEG	0	1000.4	0.314	3.40	0	0.025	0.429	0.700	0.98	0.82	0.65	0.59	9.5	0.888	0.979	12.0	0.656	0.704	14.0	0.750	0.766	16.5	0.975	0.952	19.0	0.728	0.858
107	80\	JL1580	.SEG	0	1002.8	0.310	3.00	0	0.025	0.448	0.500	0.93	0.72	0.61	0.61	9.0	1.212	1.336	12.5	0.764	0.779	15.0	1.074	1.012	18.0	1.084	1.033	19.0	1.185	1.292
108	80\	JL1680	.SEG	0	998.7	0.310	3.00	0	0.025	0.451	0.500	0.90	0.72	0.56	0.52	9.0	1.090	1.001	12.0	1.094	0.984	15.0	1.067	0.975	18.0	1.000	1.000	19.0	1.000	1.000
109	80\	JL1780	.SEG	0	996.7	0.310	3.00	0	0.025	0.452	0.550	0.85	0.74	0.58	0.54	9.0	1.175	1.005	12.0	1.085	0.949	15.0	1.052	0.937	18.0	1.048	0.947	19.0	1.069	1.000
110	80\	JL2580	.SEG	0	1002.5	0.307	3.00	0	0.025	0.465	0.700	0.97	0.72	0.55	0.52	9.5	0.782	0.964	12.0	1.032	0.993	15.0	0.997	0.977	18.0	0.977	1.013	19.0	1.065	1.205
111	80\	JL3080	.SEG	0	999.1	0.305	3.00	0	0.025	0.473	0.700	0.98	0.65	0.54	0.50	10.5	1.061	1.001	12.0	0.902	0.836	15.0	1.043	0.981	18.0	1.016	0.969	19.0	1.172	1.198
112	80\	AU0180	.SEG	0	999.8	0.305	3.00	0	0.025	0.477	0.600	0.90	0.63	0.48	0.47	9.0	1.080	1.000	12.0	1.084	0.990	15.0	1.039	0.962	18.0	1.030	0.960	19.0	1.000	1.000

	A	B	C	K	L	M	N	O	P	Q	R	T	U	V	W	X	Y	Z	AA	AB	AC	AD	AE	AF	AG	AH	AI	AJ	AK	AL
113	80\	AU0880	.SEG	0	1000.4	0.302	3.00	0	0.025	0.488	0.600	0.94	0.61	0.49	0.43	9.0	1.100	1.050	12.0	1.101	0.997	15.0	1.085	1.031	18.0	1.095	1.037	19.0	1.190	1.060
114	80\	AU0980	.SEG	0	997.7	0.302	3.00	0	0.025	0.490	0.650	0.94	0.61	0.49	0.51	9.0	1.070	1.005	12.1	1.078	0.956	16.0	0.779	0.762	18.0	1.098	0.984	19.0	1.087	1.131
115	80\	AU1180	.SEG	0	1000.1	0.301	3.00	0	0.025	0.493	0.600	0.91	0.72	0.56	0.46	10.1	1.218	1.094	13.0	0.850	0.838	15.0	1.012	0.951	18.0	1.022	0.956	19.0	1.130	1.283
116	80\	AU1280	.SEG	0	998.7	0.301	3.00	0	0.025	0.494	0.600	0.90	0.72	0.54	0.47	9.0	1.045	1.057	12.0	1.020	0.946	13.0	0.923	0.878	16.5	0.545	0.561	19.0	1.081	1.093
117	80\	AU1380	.SEG	0	1000.1	0.301	3.00	0	0.025	0.496	0.700	0.90	0.63	0.45	0.42	9.0	1.191	1.004	12.0	1.118	0.982	15.0	1.070	0.968	18.0	1.102	0.989	19.0	1.217	1.030
118	80\	AU1480	.SEG	0	1000.4	0.300	3.00	0	0.025	0.498	0.600	0.97	0.65	0.47	0.44	9.0	1.050	1.001	11.5	1.057	0.949	14.5	0.798	0.749	16.0	0.937	0.864	19.0	0.804	0.821
119	80\	AU2680	.SEG	0	1004.5	0.293	2.30	0	0.025	0.432	0.600	0.98	0.60	0.42	0.33	9.0	1.130	1.070	12.0	1.137	1.008	15.0	1.085	0.994	18.0	1.079	0.961	19.0	1.050	1.008
120	80\	AU2780	.SEG	0	1005.8	0.293	2.50	0	0.025	0.426	0.600	0.98	0.63	0.43	0.38	9.0	1.160	1.080	12.0	1.101	0.990	14.0	0.931	0.897	18.0	1.071	0.997	19.0	1.050	1.007
121	81\	JN1581	.SEG	6	1001.1	0.330	3.00	0	0.025	0.350	0.700	0.88	0.57	0.44	0.43	9.0	1.000	1.000	12.0	1.000	1.000	15.0	1.000	1.000	18.0	1.000	1.000	19.0	1.000	1.000
122	81\	JN1681	.SEG	6	999.1	0.329	3.02	0	0.025	0.353	0.700	0.82	0.50	0.40	0.34	9.0	1.000	1.000	12.0	1.000	1.000	15.0	1.000	1.000	18.0	1.000	1.000	19.0	1.000	1.000
123	81\	JN1781	.SEG	6	998.4	0.329	3.03	0	0.025	0.357	0.700	0.82	0.55	0.47	0.44	9.0	1.000	1.000	12.0	1.000	1.000	15.0	1.000	1.000	18.0	1.000	1.000	19.0	1.000	1.000
124	81\	JN1881	.SEG	6	1001.4	0.328	3.05	0	0.025	0.360	0.700	0.82	0.71	0.58	0.79	9.0	1.000	1.000	12.0	1.000	1.000	15.0	1.000	1.000	18.0	1.000	1.000	19.0	1.000	1.000
125	81\	JN1981	.SEG	6	1000.1	0.327	3.07	0	0.025	0.363	0.700	0.97	0.74	0.61	0.82	9.0	1.000	1.000	12.0	1.000	1.000	15.0	1.000	1.000	18.0	1.000	1.000	19.0	1.000	1.000
126	81\	JN2181	.SEG	6	997.7	0.326	3.10	0	0.025	0.370	0.700	0.90	0.59	0.48	0.49	9.0	1.000	1.000	12.0	1.000	1.000	15.0	1.000	1.000	18.0	1.000	1.000	19.0	1.000	1.000
127	81\	JN2281	.SEG	6	995.0	0.325	3.12	0	0.025	0.373	0.700	0.88	0.54	0.36	0.31	9.0	1.000	1.000	12.0	1.000	1.000	15.0	1.000	1.000	18.0	1.000	1.000	19.0	1.000	1.000
128	81\	JN2381	.SEG	6	1000.1	0.325	2.90	0	0.025	0.377	0.700	0.58	0.46	0.38	0.44	9.0	1.000	1.000	12.0	1.000	1.000	15.0	1.000	1.000	18.0	1.000	1.000	19.0	1.000	1.000
129	81\	JN2481	.SEG	6	1004.2	0.324	3.00	0	0.025	0.380	0.700	0.76	0.45	0.37	0.36	9.0	1.000	1.000	12.0	1.000	1.000	15.0	1.000	1.000	18.0	1.000	1.000	19.0	1.000	1.000
130	81\	JN2581	.SEG	6	1001.1	0.323	3.17	0	0.025	0.383	0.700	0.85	0.59	0.44	0.39	9.0	1.000	1.000	12.0	1.000	1.000	15.0	1.000	1.000	18.0	1.000	1.000	19.0	1.000	1.000
131	81\	JN2981	.SEG	6	1008.9	0.321	2.50	0	0.025	0.397	0.700	0.75	0.30	0.28	0.26	9.0	1.000	1.000	12.0	1.000	1.000	15.0	1.000	1.000	18.0	1.000	1.000	19.0	1.000	1.000
132	81\	JL0681	.SEG	6	998.4	0.316	3.00	0	0.025	0.419	0.700	0.94	0.85	0.72	0.70	9.0	1.000	1.000	12.0	1.000	1.000	15.0	1.000	1.000	18.0	1.000	1.000	19.0	1.000	1.000
133	81\	JL0881	.SEG	6	1003.8	0.315	3.00	0	0.025	0.426	0.700	0.97	0.65	0.44	0.47	9.0	0.900	0.900	12.0	0.900	0.900	15.0	0.900	0.900	18.0	0.900	0.900	19.0	0.900	0.900
134	81\	JL0981	.SEG	6	1001.4	0.314	3.00	0	0.025	0.429	0.700	0.94	0.61	0.49	0.51	9.0	1.000	1.000	12.0	1.000	1.000	15.0	1.000	1.000	18.0	1.000	1.000	19.0	1.000	1.000
135	81\	JL1581	.SEG	6	997.7	0.310	3.00	0	0.025	0.448	0.750	0.68	0.44	0.43	0.42	9.0	1.000	1.000	12.0	1.000	1.000	15.0	1.000	1.000	18.0	1.000	1.000	19.0	1.000	1.000
136	81\	JL1881	.SEG	6	1000.1	0.309	3.50	0	0.025	0.454	0.700	0.90	0.59	0.52	0.48	9.0	1.040	1.006	12.0	1.043	1.002	15.0	0.992	1.003	18.0	1.070	1.040	19.0	1.000	1.000
137	81\	JL2081	.SEG	6	997.7	0.309	4.00	0	0.025	0.457	0.700	0.94	0.75	0.56	0.53	9.0	1.000	1.000	12.0	1.000	1.000	15.0	1.000	1.000	18.0	1.000	1.000	19.0	1.000	1.000
138	81\	JL2181	.SEG	6	993.0	0.308	3.60	0	0.025	0.459	0.700	0.98	0.80	0.68	0.52	9.0	0.966	1.029	12.0	0.982	0.904	15.0	0.822	0.953	18.0	0.814	0.861	19.0	1.000	1.000
139	81\	JL2281	.SEG	6	996.7	0.308	3.30	0	0.025	0.461	0.700	0.82	0.61	0.50	0.47	9.0	1.190	1.150	12.5	1.126	1.049	14.5	0.805	0.823	18.0	0.955	0.910	19.0	1.000	1.000
140	81\	JL2381	.SEG	6	1002.5	0.308	3.50	0	0.025	0.462	0.700	0.93	0.63	0.52	0.45	9.0	0.877	0.950	12.5	0.914	0.856	13.8	0.591	0.804	16.0	1.055	0.984	19.0	1.000	1.000
141	81\	JL3181	.SEG	6	1008.6	0.305	3.50	0	0.025	0.460	0.700	0.73	0.60	0.43	0.43	9.0	1.000	1.000	12.0	1.000	0.993	15.0	1.000	1.005	18.0	1.000	0.995	19.0	1.000	1.043
142	81\	AU0581	.SEG	6	1001.8	0.303	3.40	0	0.025	0.483	0.600	0.97	0.70	0.58	0.51	9.0	1.150	1.034	11.5	1.083	0.942	15.0	0.921	0.843	18.0	1.063	0.991	19.0	1.111	1.050
143	81\	AU1381	.SEG	6	1003.8	0.301	3.30	0	0.025	0.496	0.450	0.97	0.77	0.57	0.59	9.0	1.140	0.981	12.0	1.068	0.919	15.0	0.994	0.900	18.0	0.905	0.826	19.0	1.000	1.000
144	81\	AU1481	.SEG	6	1004.2	0.300	2.90	0	0.025	0.498	0.450	0.97	0.79	0.48	0.46	9.0	1.182	1.050	12.0	1.112	0.945	15.0	1.043	0.887	18.0	0.996	0.906	19.0	1.150	1.190
145	81\	AU2681	.SEG	6	1004.5	0.293	2.80	0	0.025	0.432	0.600	0.97	0.73	0.62	0.49	9.5	1.125	1.063	12.5	1.126	1.007	15.0	1.049	0.946	18.0	0.928	0.919	19.0	1.000	1.020
146	81\	AU2781	.SEG	6	1003.5	0.293	2.90	0	0.025	0.426	0.600	0.98	0.69	0.54	0.49	9.0	1.140	1.020	11.5	1.080	1.000	12.0	0.900	0.877	14.5	1.107	0.971	19.0	1.063	0.988
147	81\	AU3181	.SEG	6	1000.8	0.290	3.20	0	0.025	0.400	0.600	0.98	0.69	0.55	0.53	9.0	0.941	1.068	12.5	1.092	0.967	13.0	1.038	0.927	15.0	1.058	0.963	19.0	1.030	1.004
148	81\	ST0281	.SEG	6	999.1	0.289	2.80	0	0.025	0.387	0.600	0.97	0.69	0.61	0.61	9.0	1.130	1.090	12.7	1.091	0.964	14.7	0.882	0.900	18.0	0.754	0.804	19.0	1.000	1.000
149	81\	ST0381	.SEG	6	1000.4	0.288	2.80	0	0.025	0.380	0.600	0.97	0.79	0.65	0.63	9.0	0.912	0.930	12.0	1.008	0.902	14.2	0.719	0.687	18.0	0.352	0.419	19.0	1.000	1.000
150	81\	ST0981	.SEG	6	997.0	0.284	2.40	0	0.025	0.340	0.700	0.93	0.66	0.54	0.40	9.0	1.020	1.080	12.0	1.069	1.007	14.0	0.840	0.798	18.0	1.030	0.909	19.0	1.000	1.000
151	81\	ST1081	.SEG	6	1000.8	0.283	2.40	0	0.025	0.333	0.600	0.87	0.59	0.40	0.44	9.0	1.180	1.090	12.0	1.100	1.005	15.0	1.094	0.976	18.0	1.131	1.000	19.0	1.000	1.000
152	81\	ST1281	.SEG	6	1001.1	0.282	2.50	0	0.025	0.320	0.600	0.90	0.64	0.53	0.46	9.0	1.190	1.190	12.0	0.992	0.935	15.0	0.961	0.855	18.0	0.959	0.897	19.0	2.000	1.532
153	81\	ST1481	.SEG	6	998.1	0.281	2.70	0	0.025	0.307	0.400	0.93	0.74	0.55	0.48	9.0	1.120	1.130	13.0	1.057	0.992	15.0	0.910	0.851	18.0	1.080	1.039	19.0	1.000	1.000
154	81\	ST1781	.SEG	6	1001.1	0.279	2.80	0	0.025	0.292	0.600	0.97	0.84	0.73	0.69	9.0	1.000	1.000	12.0	1.000	1.000	15.0	1.000	1.000	18.0	1.000	1.000	19.0	1.000	1.000
155	81\	ST2081	.SEG	6	999.8	0.278	2.60	0	0.025	0.280	0.500	0.93	0.59	0.48	0.36	9.0	1.180	1.100	12.0	1.142	1.086	15.0	1.093	0.988	18.0	1.107	1.000	19.0	1.000	1.000
156	81\	ST2481	.SEG	6	1007.9	0.277	2.00	0	0.025	0.264	0.700	0.86	0.50	0.44	0.37	9.0	1.100	1.090	12.0	1.100	0.997	15.0	1.078	0.977	18.0	1.111	1.004	19.0	1.000	1.000
157	81\	ST2881	.SEG	6	1003.1	0.276	2.29	0	0.025	0.248	0.700	0.93	0.45	0.33	0.28	9.0	1.150	1.100	12.0	1.087	1.012	15.0	1.080	0.990	18.0	1.096	0.999	19.0	1.000	1.000
158	81\	ST2981	.SEG	6	1006.2	0.275	2.00	0	0.025	0.244	0.700	0.28	0.52	0.39	0.32	9.0	1.140	1.090	12.0	1.109	1.019	15.0	1.058	0.987	18.0	1.083	1.020	19.0	1.000	1.000
159	81\	OC0181	.SEG	6	997.7	0.275	2.60	0	0.025	0.236	0.650	0.93	0.76	0.51	0.39	9.0	1.040	1.050	12.0	1.009	0.937	15.0	0.942	0.860	18.0	0.909	0.922	19.0	1.000	1.000
160	81\	OC1481	.SEG	6	1011.9	0.270	1.70	0	0.025	0.186	0.700	0.83	0.60	0.45	0.42	9.0	1.120	1.040	12.0	1.095	1.029	15.0	1.048	1.019	18.0	1.007	1.040	19.0	1.000	1.000
161	81\	NO1381	.SEG	6	1012.6	0.279	1.67	2	0.025	0.143	0.880	0.59	0.47	0.31	0.22	9.0	1.000	1.000	12.0	1.000	1.000	15.0	1.000	1.000	18.0	1.000	1.000	19.0	1.000	1.000
162	81\	NO1581	.SEG	6	1005.8	0.280	1.64	2	0.025	0.140	0.880	0.64	0.33	0.22	0.21	9.0	1.000	1.000	12.0	1.000	1.000	15.0	1.000	1.000	18.0	1.000	1.000	19.0	0.726	0.766
163	81\	NO2281	.SEG	6	1006.6	0.282	1.51	2	0.025	0.131	0.880	0.88	0.57	0.48	0.39	9.0	1.000	1.000	12.0	1.000	1.000	15.0	1.000	1.000	18.0	1.000	1.000	19.0	1.000	1.000
164	82\	JN0682	.SEG	10	996.7	0.333	2.60	2	0.100	0.335	0.880	0.61	0.45	0.44	0.50	10.5	1.122	1.039	13.0	1.144	1.048	16.0	0.941	0.933	18.0	0.415	0.486	19.0	0.994	0.924
165	82\	JN0882	.SEG	10	999.1	0.332	2.80	0	0.100	0.338	0.600	0.81	0.54	0.49	0.48	9.0	1.000	1.040	13.0	1.022	0.977	15.0	0.931	0.889	18.0	0.850	0.875	19.0	0.667	0.836
166	82\	JN0982	.SEG	10	999.8	0.332	2.80	2	0.100	0.340	0.880	0.90	0.76	0.65	0.60	9.0	0.884	0.947	12.5	0.992	0.927	15.7	0.640	0.652	18.0	0.879	0.873	19.0	0.750	0.925
167	82\	JN1482	.SEG	10	1001.1	0.330	2.50	0	0.100	0.348	0.700	0.93	0.71	0.54	0.49	9.0	0.770	0.923	13.0	0.923	0.916	15.0	0.891	0.866	18.0	0.740	0.764	19.0	0.471	0.606

	A	B	C	K	L	M	N	O	P	Q	R	T	U	V	W	X	Y	Z	AA	AB	AC	AD	AE	AF	AG	AH	AI	AJ	AK	AL
168	82\	JN1582	SEG	10	1001.4	0.330	3.20	0	0.100	0.350	0.600	0.87	0.63	0.49	0.49	9.0	1.029	1.070	11.8	1.035	1.005	13.8	0.743	0.712	15.5	0.779	0.764	19.0	0.447	0.633
169	82\	JN1682	SEG	10	997.7	0.329	3.02	0	0.100	0.353	0.700	0.87	0.74	0.57	0.74	9.0	0.813	0.878	13.0	0.854	0.802	15.0	0.561	0.598	17.0	0.403	0.466	19.0	0.599	0.925
170	82\	JN2582	SEG	10	1003.8	0.323	3.10	0	0.100	0.383	0.700	0.78	0.64	0.55	0.51	9.0	0.694	0.764	11.8	0.617	0.637	14.0	0.853	0.809	16.0	0.780	0.756	19.0	0.725	0.778
171	82\	JN2782	SEG	10	1002.5	0.322	3.10	0	0.100	0.390	0.700	0.87	0.74	0.55	0.57	10.0	0.426	0.522	13.0	0.880	0.875	15.0	0.772	0.810	18.0	0.916	0.936	19.0	1.000	1.000
172	82\	JN2982	SEG	10	993.0	0.321	3.00	2	0.100	0.397	0.880	0.90	0.67	0.59	0.79	8.6	0.582	0.676	11.8	0.958	0.859	13.6	0.818	0.758	15.0	0.617	0.649	19.0	0.812	0.785
173	82\	JN3082	SEG	10	994.0	0.320	2.50	0	0.100	0.400	0.700	0.79	0.57	0.48	0.48	9.0	1.030	1.060	11.5	1.130	1.000	13.0	0.994	1.034	14.2	0.974	0.972	19.0	0.983	0.990
174	82\	JL0182	SEG	10	1001.8	0.319	2.80	0	0.100	0.403	0.700	0.73	0.48	0.45	0.39	9.0	1.087	1.091	12.0	1.077	1.003	15.0	1.069	1.027	18.0	1.074	1.025	19.0	1.000	1.000
175	82\	JL0382	SEG	10	998.1	0.318	3.10	0	0.100	0.410	0.700	0.82	0.67	0.57	0.54	9.0	0.909	0.900	12.0	0.972	0.900	15.0	0.997	0.920	18.0	1.015	0.942	19.0	0.748	0.896
176	82\	JL0682	SEG	10	1006.5	0.316	3.35	0	0.100	0.419	0.700	0.84	0.62	0.49	0.51	9.5	0.564	0.732	11.1	0.596	0.670	15.0	1.065	1.037	18.0	1.045	1.039	19.0	1.000	1.000
177	82\	JL0882	SEG	10	1002.8	0.315	3.30	0	0.100	0.426	0.700	0.87	0.61	0.52	0.48	9.0	1.125	1.099	12.5	1.093	1.012	13.5	1.024	0.986	18.0	1.000	1.000	19.0	1.175	1.083
178	82\	JL1582	SEG	10	1005.5	0.310	3.30	0	0.100	0.448	0.700	0.87	0.85	0.74	0.65	9.0	0.219	0.313	12.0	0.214	0.307	15.0	0.434	0.537	18.0	0.668	0.751	19.0	1.000	0.966
179	82\	JL1782	SEG	10	1004.8	0.310	2.80	0	0.100	0.452	0.700	0.90	0.65	0.55	0.54	10.0	1.080	1.050	12.0	0.927	0.911	15.0	0.924	0.919	18.0	1.090	1.046	19.0	0.695	0.848
180	82\	JL1982	SEG	10	998.4	0.309	2.90	0	0.100	0.456	0.700	0.90	0.67	0.57	0.56	9.0	0.864	0.827	12.5	1.012	0.964	15.0	0.882	0.811	18.0	1.030	0.977	19.0	0.667	0.740
181	82\	JL2682	SEG	10	1002.8	0.307	3.00	0	0.100	0.467	0.700	0.93	0.63	0.54	0.47	9.0	0.944	1.004	12.1	1.057	0.953	13.8	1.021	0.931	15.8	0.900	0.848	19.0	0.969	0.941
182	82\	JL2782	SEG	10	1001.8	0.306	3.50	0	0.100	0.469	0.500	0.90	0.61	0.47	0.45	9.0	1.093	1.070	12.0	0.980	0.923	16.0	1.023	0.943	18.0	0.857	0.845	19.0	0.923	1.154
183	82\	JL2882	SEG	10	998.1	0.306	3.50	0	0.100	0.470	0.600	0.87	0.65	0.61	0.77	9.0	1.120	1.009	12.0	1.012	0.922	13.5	0.733	0.756	16.0	0.169	0.228	19.0	0.333	0.403
184	82\	AU0282	SEG	10	998.4	0.304	3.50	0	0.100	0.478	0.500	0.90	0.56	0.51	0.46	9.0	1.175	1.250	12.0	1.056	1.000	13.5	0.940	0.895	15.0	0.730	0.771	19.0	1.143	1.014
185	82\	AU0382	SEG	10	998.7	0.304	3.10	0	0.100	0.480	0.480	0.81	0.67	0.53	0.50	9.0	1.103	1.240	12.5	1.097	1.050	14.0	0.940	0.881	16.0	0.807	0.831	19.0	0.941	0.911
186	82\	AU0682	SEG	10	1004.5	0.303	3.10	0	0.100	0.485	0.500	0.87	0.65	0.50	0.53	9.0	1.000	1.109	12.0	1.124	1.059	15.0	1.065	1.005	18.0	0.883	0.883	19.0	1.000	1.000
187	82\	AU0782	SEG	10	1004.5	0.303	3.40	0	0.100	0.486	0.600	0.90	0.70	0.55	0.54	9.0	1.116	1.080	12.5	1.032	0.970	14.5	0.894	0.860	16.2	1.032	0.966	19.0	0.455	0.564
188	82\	AU2082	SEG	10	1002.8	0.297	2.90	0	0.100	0.471	0.700	0.87	0.67	0.53	0.52	9.0	1.143	1.050	12.0	1.100	1.030	15.0	1.072	0.968	18.0	0.252	0.367	19.0	1.000	1.000
189	82\	AU2282	SEG	10	1002.5	0.296	2.90	0	0.100	0.458	0.800	0.81	0.64	0.52	0.51	9.0	0.844	0.748	13.0	1.058	0.942	14.8	0.939	0.909	18.0	0.902	0.847	19.0	1.000	1.000
190	82\	AU2382	SEG	10	1001.1	0.295	3.00	0	0.100	0.462	0.600	0.90	0.85	0.56	0.50	10.0	0.237	0.359	13.0	0.925	0.936	14.5	1.006	0.977	16.0	0.766	0.813	19.0	1.000	1.042
191	82\	AU2482	SEG	10	1000.4	0.295	2.90	0	0.100	0.448	0.600	0.84	0.56	0.47	0.47	9.8	1.300	1.206	12.0	0.821	0.827	15.0	0.982	0.946	18.0	0.839	0.878	19.0	0.750	1.000
192	82\	AU2782	SEG	10	1000.1	0.293	3.13	0	0.100	0.426	0.600	0.84	0.69	0.61	0.53	9.2	1.222	1.214	13.0	0.766	0.798	15.0	0.761	0.776	18.0	0.765	0.812	19.0	1.000	1.000
193	82\	ST0382	SEG	10	997.0	0.286	2.50	0	0.100	0.360	0.700	0.87	0.61	0.48	0.59	10.5	1.131	1.061	11.5	0.747	0.794	13.0	0.951	0.929	15.0	0.821	0.846	19.0	0.462	0.818
194	82\	ST0482	SEG	10	1002.8	0.287	2.20	0	0.100	0.373	0.700	0.68	0.43	0.38	0.39	9.0	1.027	1.041	12.0	1.024	0.975	15.0	1.021	0.994	18.0	1.065	1.040	19.0	1.136	0.846
195	82\	ST0582	SEG	10	1006.5	0.287	2.83	0	0.100	0.367	0.700	0.81	0.55	0.47	0.43	9.0	1.417	1.251	12.8	0.904	0.890	15.0	0.795	0.815	18.0	1.029	1.016	19.0	1.000	1.000
196	82\	ST0682	SEG	10	1006.5	0.286	2.50	0	0.100	0.360	0.700	0.81	0.53	0.42	0.37	9.0	1.300	1.100	12.0	1.100	1.064	15.0	1.064	1.048	18.0	1.074	1.076	19.0	1.000	1.304
197	82\	ST0782	SEG	10	1006.2	0.285	2.40	0	0.100	0.353	0.700	0.84	0.50	0.40	0.38	9.0	1.090	1.020	12.0	1.091	1.000	15.0	1.042	0.985	18.0	1.036	1.005	19.0	1.000	1.000
198	82\	ST0882	SEG	10	1007.9	0.285	2.40	0	0.100	0.347	0.700	0.90	0.71	0.54	0.49	9.0	0.343	0.447	12.0	0.424	0.495	15.0	0.911	0.913	18.0	0.843	0.869	19.0	1.000	1.000
199	82\	ST1282	SEG	10	1006.2	0.282	2.40	0	0.100	0.320	0.700	0.87	0.60	0.43	0.39	9.0	1.081	1.116	12.0	1.087	1.026	15.0	0.997	0.996	18.0	0.965	1.014	19.0	1.000	1.000
200	82\	ST1382	SEG	10	1007.2	0.281	2.57	0	0.100	0.313	0.700	0.90	0.58	0.53	0.45	9.0	0.969	1.128	12.0	1.030	1.025	15.0	1.034	1.019	18.0	0.966	0.994	19.0	1.000	1.000
201	82\	ST1482	SEG	10	1004.2	0.281	2.53	0	0.100	0.307	0.700	0.87	0.65	0.53	0.53	9.0	0.719	0.805	11.5	0.716	0.764	13.5	0.523	0.579	16.0	0.759	0.824	19.0	0.618	0.640
202	82\	ST1682	SEG	10	998.4	0.280	2.40	0	0.100	0.296	0.700	0.90	0.57	0.44	0.47	9.0	1.129	1.050	13.0	1.000	0.941	15.0	0.875	0.862	18.0	0.804	0.864	19.0	1.000	1.255
203	82\	ST1782	SEG	10	1000.8	0.279	2.40	0	0.100	0.292	0.700	0.68	0.55	0.47	0.47	9.8	0.419	0.529	12.5	0.725	0.747	13.5	0.571	0.597	15.0	0.818	0.823	19.0	0.926	0.982
204	82\	ST1882	SEG	10	997.0	0.279	3.00	0	0.100	0.288	0.600	0.97	0.87	0.52	0.43	10.1	0.391	0.555	11.0	1.000	0.987	15.0	0.979	0.953	18.0	0.801	0.921	19.0	1.000	1.250
205	82\	ST2482	SEG	10	1003.5	0.277	2.36	0	0.100	0.264	0.700	0.86	0.42	0.34	0.37	9.0	1.000	1.000	12.0	1.000	1.000	15.0	1.000	1.000	18.0	1.000	1.000	19.0	1.000	1.000
206	82\	ST2982	SEG	10	1006.5	0.275	2.28	2	0.100	0.244	0.880	0.90	0.64	0.50	0.50	9.0	0.935	1.050	12.8	0.953	0.923	15.0	0.411	0.489	18.0	0.614	0.696	19.0	1.000	1.000
207	82\	OC0182	SEG	10	1002.5	0.275	2.25	2	0.100	0.236	0.880	0.90	0.68	0.52	0.49	9.0	0.933	1.050	12.0	0.991	0.966	15.0	1.000	0.951	18.0	0.901	0.934	19.0	1.000	1.000
208	82\	OC0382	SEG	10	1002.1	0.274	3.00	0	0.100	0.228	0.700	0.93	0.61	0.50	0.54	9.0	0.917	1.008	12.0	0.944	0.935	15.0	0.943	0.926	18.0	0.907	0.949	19.0	1.000	1.000
209	82\	OC0682	SEG	10	1007.9	0.273	2.80	0	0.100	0.217	0.700	0.93	0.74	0.49	0.48	9.0	0.957	1.030	12.0	1.000	0.973	15.0	0.929	0.937	18.0	0.870	0.952	19.0	0.684	0.507
210	82\	OC0982	SEG	10	1001.8	0.272	2.14	0	0.100	0.209	0.700	0.87	0.64	0.74	0.87	9.0	0.909	1.107	12.0	0.199	0.283	15.0	0.151	0.243	18.0	0.148	0.256	19.0	1.000	1.000
211	82\	OC2782	SEG	10	1010.2	0.274	1.50	2	0.100	0.165	0.880	0.68	0.60	0.32	0.30	9.0	1.070	1.080	12.0	1.047	1.031	15.0	1.011	1.005	18.0	1.000	1.142	19.0	1.000	1.000
212	82\	OC2882	SEG	10	1009.6	0.274	1.87	2	0.100	0.164	0.880	0.79	0.62	0.31	0.30	9.0	1.040	1.040	12.0	1.043	0.997	16.0	0.994	0.968	18.0	1.000	1.030	19.0	1.000	1.000
213	82\	NO1182	SEG	10	1010.6	0.279	1.60	1	0.100	0.200	0.946	0.93	0.64	0.39	0.45	9.0	1.100	1.040	12.0	0.933	0.943	16.0	0.891	0.923	18.0	0.452	0.684	19.0	1.000	1.000
214	82\	NO1582	SEG	10	1008.2	0.280	1.50	1	0.100	0.200	0.946	0.89	0.56	0.41	0.33	9.0	1.170	1.060	12.0	1.101	1.003	16.0	1.033	1.016	18.0	1.020	1.004	19.0	1.000	1.000
215	82\	DE0782	SEG	10	1013.6	0.287	1.30	1	0.100	0.200	0.946	0.71	0.63	0.51	0.47	9.0	1.000	1.003	12.0	1.048	0.992	15.0	1.044	1.049	18.0	0.939	1.070	19.0	1.000	1.000
216	83\	JN1483	SEG	0	1001.8	0.330	2.99	0	0.100	0.348	0.600	0.78	0.56	0.45	0.50	9.0	0.900	1.010	12.0	0.848	0.840	13.0	0.886	0.852	15.2	0.926	0.894	19.0	0.421	0.487
217	83\	JN2783	SEG	0	1002.8	0.322	3.00	2	0.100	0.390	0.880	0.87	0.74	0.68	0.59	9.0	0.671	0.573	12.0	0.760	0.706	15.0	0.799	0.751	18.0	0.585	0.592	19.0	1.000	1.000
218	83\	JL0283	SEG	0	1001.8	0.319	3.00	2	0.100	0.407	0.880	0.94	0.72	0.54	0.52	9.0	0.679	0.774	13.1	1.041	0.929	15.2	0.884	0.846	17.0	0.645	0.651	19.0	0.956	0.854
219	83\	JL0883	SEG	0	1008.2	0.315	3.00	2	0.100	0.426	0.880	0.81	0.52	0.45	0.42	9.0	1.078	1.009	12.0	1.111	0.966	15.0	1.072	1.012	18.0	1.071	0.983	19.0	1.000	1.000
220	83\	JL0983	SEG	0	1001.4	0.314	2.50	2	0.100	0.429	0.880	0.78	0.49	0.35	0.34	9.0	1.039	0.973	12.0	1.064	0.928	15.0	1.041	0.973	18.0	1.060	0.966	19.0	1.000	1.000
221	83\	JL1583	SEG	0	998.7	0.310	2.50	0	0.100	0.448	0.600	0.74	0.53	0.44	0.38	9.0	1.256	1.128	12.0	1.129	1.009	15.0	1.042	0.983	18.0	1.074	1.003	19.0	1.000	1.000
222	83\	JL1783	SEG	0	998.1	0.310	2.60	0	0.100	0.452	0.600	0.79	0.57	0.40	0.41	9.0	1.105	1.037	13.1	1.033	0.918	15.1	0.863	0.802	18.0	0.800	0.760	19.0	1.000	1.000

	A	B	C	K	L	M	N	O	P	Q	R	T	U	V	W	X	Y	Z	AA	AB	AC	AD	AE	AF	AG	AH	AI	AJ	AK	AL
223	83\	JL2183	.SEG	0	999.1	0.306	3.00	0	0.100	0.459	0.650	0.82	0.54	0.46	0.37	9.7	1.190	1.110	12.0	1.037	0.912	15.0	0.956	0.865	18.0	0.752	0.672	19.0	1.000	1.000
224	83\	JL2483	.SEG	0	994.3	0.307	3.00	0	0.100	0.464	0.650	0.87	0.77	0.44	0.61	9.0	0.895	0.856	12.5	1.009	0.894	14.2	0.935	0.894	18.0	0.600	0.529	19.0	1.000	1.000
225	83\	JL2683	.SEG	0	999.1	0.307	3.00	2	0.100	0.467	0.860	0.93	0.67	0.51	0.46	9.0	1.030	1.004	12.0	1.017	0.910	15.0	0.963	0.890	18.0	0.962	0.863	19.0	1.000	1.000
226	83\	JL2783	.SEG	0	1004.5	0.306	3.00	1	0.100	0.469	0.946	0.90	0.51	0.40	0.40	9.0	0.971	0.867	12.0	1.065	0.882	15.0	1.040	0.944	18.0	0.266	0.898	19.0	1.000	1.000
227	83\	JL2983	.SEG	0	1008.9	0.306	3.00	1	0.100	0.472	0.946	0.90	0.58	0.39	0.36	9.0	1.044	0.906	12.0	1.089	0.982	14.0	1.058	0.990	18.0	1.036	0.984	19.0	1.033	0.980
228	83\	JL3183	.SEG	0	1004.8	0.305	3.00	1	0.100	0.475	0.946	0.82	0.59	0.50	0.44	9.0	1.060	0.902	11.2	1.067	0.907	13.0	0.901	0.836	15.0	0.600	0.614	19.0	0.853	0.668
229	83\	AU0183	.SEG	0	1004.8	0.305	3.00	1	0.100	0.477	0.946	0.82	0.63	0.44	0.41	9.0	1.015	0.861	11.1	1.073	0.893	13.1	0.866	0.803	15.1	0.886	0.841	19.0	0.843	0.842
230	83\	AU0483	.SEG	0	1003.8	0.304	3.00	2	0.100	0.482	0.860	0.97	0.90	0.57	0.52	10.5	0.397	0.486	11.0	0.909	0.858	15.0	1.016	1.013	18.0	0.916	0.945	19.0	1.000	1.000
231	83\	AU0683	.SEG	0	1000.4	0.303	3.00	2	0.100	0.485	0.860	0.93	0.70	0.50	0.82	9.0	0.950	0.828	12.8	0.964	0.847	14.0	0.176	0.248	15.5	0.370	0.400	19.0	0.492	0.486
232	83\	AU1183	.SEG	0	996.4	0.301	3.00	0	0.100	0.493	0.800	0.90	0.61	0.37	0.41	9.0	0.942	0.844	12.0	0.993	0.828	15.0	0.959	0.837	18.0	0.778	0.698	19.0	1.000	1.000
233	83\	AU1383	.SEG	0	1001.1	0.301	3.00	1	0.100	0.496	0.946	0.84	0.57	0.48	0.48	9.0	1.102	0.926	12.0	1.089	0.927	15.0	0.828	0.803	18.0	0.831	0.826	19.0	1.000	1.000
234	83\	AU1483	.SEG	0	1004.8	0.300	3.00	2	0.100	0.498	0.860	0.87	0.59	0.48	0.43	9.0	1.000	1.000	12.0	1.000	1.000	15.0	1.000	1.000	18.0	1.000	1.000	19.0	1.000	1.182
235	83\	AU1883	.SEG	0	1000.1	0.298	3.00	2	0.100	0.484	0.860	0.93	0.74	0.52	0.38	9.5	0.671	0.619	12.0	0.449	0.480	14.0	0.524	0.543	18.0	0.852	0.804	19.0	1.000	1.000
236	83\	AU1983	.SEG	0	999.4	0.298	3.00	0	0.100	0.477	0.700	0.87	0.59	0.49	0.40	9.0	1.070	0.906	12.0	0.925	0.856	14.1	1.046	0.959	18.0	0.831	0.819	19.0	1.000	1.000
237	83\	AU2283	.SEG	0	999.1	0.296	3.00	0	0.100	0.456	0.600	0.88	0.59	0.42	0.35	9.0	1.160	1.100	12.5	1.037	0.937	15.0	0.683	0.705	18.0	1.022	0.995	19.0	1.000	1.000
238	83\	AU2683	.SEG	0	1006.5	0.293	2.80	0	0.100	0.432	0.800	0.93	0.62	0.51	0.49	9.0	1.156	1.000	12.0	1.077	0.965	15.0	1.028	1.004	18.0	0.962	0.956	19.0	1.000	1.000
239	83\	ST1283	.SEG	0	998.1	0.282	2.50	0	0.100	0.320	0.600	0.84	0.55	0.41	0.37	9.0	1.250	1.140	12.2	1.027	0.994	14.1	0.966	0.925	16.2	0.836	0.881	19.0	0.478	0.504
240	83\	ST2383	.SEG	0	1009.6	0.277	2.20	1	0.100	0.300	0.946	0.86	0.43	0.32	0.32	9.0	1.108	1.100	12.0	1.085	1.015	15.0	1.033	1.017	18.0	1.000	1.001	19.0	1.000	1.000
241	83\	ST2583	.SEG	0	1014.0	0.277	2.00	1	0.100	0.300	0.946	0.86	0.45	0.39	0.38	9.0	1.111	1.100	12.0	1.111	1.026	15.0	1.052	1.069	18.0	1.039	1.097	19.0	1.000	1.000
242	83\	ST2783	.SEG	0	1006.5	0.276	2.31	1	0.100	0.252	0.946	0.83	0.54	0.45	0.41	9.0	1.057	1.060	12.0	1.021	0.986	15.0	0.975	0.984	18.0	0.921	0.970	19.0	1.000	1.000
243	83\	OC0483	.SEG	0	999.1	0.274	2.20	1	0.100	0.225	0.946	0.87	0.56	0.43	0.44	9.0	1.129	1.080	12.5	1.028	0.982	15.0	0.848	0.891	18.0	0.793	0.856	19.0	1.000	1.000
244	83\	OC0683	.SEG	0	1000.1	0.273	2.17	1	0.100	0.217	0.946	0.87	0.61	0.50	0.47	9.0	0.867	1.143	12.0	1.030	1.032	15.0	0.883	0.922	18.0	0.756	0.877	19.0	1.000	1.000
245	83\	OC0783	.SEG	0	1007.2	0.273	2.00	1	0.100	0.213	0.946	0.70	0.49	0.46	0.41	9.0	1.067	1.002	12.0	1.061	1.002	15.0	1.007	1.006	18.0	0.966	1.085	19.0	1.000	1.000
246	83\	OC1783	.SEG	0	1007.9	0.270	2.00	1	0.100	0.178	0.946	0.93	0.75	0.51	0.49	9.0	1.000	1.170	12.0	0.972	0.996	15.0	0.858	0.916	18.0	0.714	1.006	19.0	1.000	1.000
247	84\	JN2784	.SEG	0	997.7	0.322	2.50	0	0.070	0.390	0.700	0.73	0.56	0.36	0.40	9.0	1.135	1.000	13.1	1.116	0.919	15.0	0.853	0.747	18.0	0.977	0.810	19.0	1.000	1.000
248	84\	AU0484	.SEG	0	1001.8	0.304	2.60	1	0.070	0.482	0.946	0.88	0.55	0.52	0.48	9.0	1.050	0.914	13.0	1.057	0.897	14.0	0.740	0.764	16.0	0.966	0.864	19.0	0.848	0.787
249	84\	AU0584	.SEG	0	1002.5	0.303	2.60	0	0.070	0.483	0.700	0.85	0.61	0.50	0.48	9.0	1.196	1.060	13.0	1.096	0.983	15.0	1.004	0.950	18.0	1.063	0.950	19.0	1.000	1.000
250	84\	AU0684	.SEG	0	1000.8	0.303	2.60	0	0.070	0.485	0.700	0.98	0.65	0.55	0.54	9.0	1.180	1.060	13.0	1.112	1.011	15.0	1.083	1.015	18.0	1.111	1.008	19.0	1.000	1.000
251	84\	AU0784	.SEG	0	997.4	0.303	2.60	0	0.070	0.486	0.700	0.88	0.57	0.54	0.50	9.0	1.280	1.060	12.0	1.121	1.018	15.0	1.081	1.006	18.0	1.103	1.006	19.0	1.000	1.000
252	84\	AU0884	.SEG	0	996.0	0.302	2.60	0	0.070	0.488	0.600	0.94	0.63	0.57	0.57	9.0	1.250	1.100	13.0	1.073	1.001	15.0	0.931	0.883	18.0	1.044	0.971	19.0	1.000	1.000
253	84\	AU0984	.SEG	0	997.7	0.302	2.80	0	0.070	0.490	0.600	0.97	0.70	0.61	0.68	9.0	1.116	1.080	12.8	1.078	1.002	15.0	1.015	0.937	18.0	1.167	1.040	19.0	1.000	1.000
254	84\	AU2284	.SEG	0	1003.5	0.296	2.00	0	0.070	0.456	0.700	0.93	0.62	0.55	0.49	9.0	1.047	0.997	12.0	1.080	1.001	15.0	1.000	1.000	18.0	1.060	1.000	19.0	1.000	1.000
255	84\	AU2384	.SEG	0	996.7	0.295	2.20	0	0.070	0.452	0.700	0.97	0.67	0.59	0.90	9.0	1.160	1.060	11.0	1.033	0.940	13.0	0.978	0.881	14.5	0.738	0.709	19.0	0.295	0.336
256	84\	AU2584	.SEG	0	1005.5	0.294	2.10	0	0.070	0.439	0.800	0.87	0.56	0.47	0.47	9.0	0.217	0.309	12.0	1.000	0.912	15.0	1.014	0.936	18.0	0.827	0.806	19.0	1.000	1.000
257	84\	AU2784	.SEG	0	1005.2	0.293	2.10	0	0.070	0.426	0.700	0.93	0.62	0.56	0.55	9.5	0.894	0.968	12.0	1.053	0.934	15.0	0.951	0.850	18.0	0.899	0.849	19.0	1.000	1.000
258	84\	ST0184	.SEG	0	1003.1	0.289	2.00	0	0.070	0.393	0.800	0.78	0.45	0.40	0.34	9.0	1.256	1.050	12.0	1.142	0.994	15.0	1.063	0.956	18.0	1.055	0.959	19.0	1.167	1.162
259	84\	ST0284	.SEG	0	1002.5	0.289	1.90	0	0.070	0.387	0.700	0.93	0.51	0.46	0.41	9.0	1.308	1.070	12.0	1.116	0.999	15.0	1.085	0.980	18.0	1.067	0.994	19.0	1.000	1.000
260	84\	ST0384	.SEG	0	997.7	0.288	3.00	0	0.070	0.380	0.650	0.93	0.65	0.52	0.47	9.0	1.100	1.140	12.8	1.019	1.001	15.0	0.884	0.881	18.0	0.883	0.909	19.0	0.091	0.197
261	84\	ST0884	.SEG	0	1012.3	0.285	1.80	2	0.070	0.347	0.880	0.83	0.59	0.47	0.41	9.0	1.209	1.050	12.0	1.134	1.007	15.0	1.080	1.000	18.0	1.071	1.007	19.0	1.000	1.000
262	84\	ST1684	.SEG	0	1007.9	0.280	1.80	1	0.070	0.296	0.946	0.60	0.50	0.42	0.39	9.0	0.690	0.848	12.0	1.050	0.953	15.0	1.075	0.994	18.0	1.094	1.024	19.0	1.000	1.000
263	84\	ST1784	.SEG	0	1012.3	0.279	1.80	1	0.070	0.292	0.946	0.83	0.52	0.41	0.47	11.0	1.161	0.977	12.0	0.972	0.915	15.0	1.102	0.998	18.0	1.087	0.966	19.0	1.000	1.000
264	84\	ST1984	.SEG	0	1003.1	0.279	1.80	0	0.070	0.284	0.700	0.86	0.56	0.44	0.37	9.0	1.240	1.100	12.0	1.167	1.040	15.0	1.108	1.014	18.0	1.126	1.038	19.0	1.000	1.000
265	84\	ST2184	.SEG	0	1000.1	0.278	1.90	0	0.070	0.276	0.700	0.87	0.52	0.40	0.43	9.0	1.220	1.100	12.0	1.118	1.016	15.0	1.065	0.985	18.0	1.017	0.992	19.0	1.000	1.000
266	84\	ST2584	.SEG	0	1002.5	0.277	2.00	2	0.070	0.260	0.880	0.97	0.76	0.53	0.48	9.0	0.600	0.926	12.0	0.643	0.688	15.0	0.635	0.668	18.0	0.964	0.968	19.0	1.000	0.041
267	84\	OC0484	.SEG	36	1004.5	0.274	1.80	0	0.070	0.225	0.800	0.83	0.53	0.40	0.40	9.0	1.214	1.050	12.0	1.079	0.976	15.0	1.052	0.975	18.0	0.981	1.024	19.0	1.000	1.000
268	84\	OC0584	.SEG	36	1007.2	0.273	2.00	0	0.070	0.221	0.700	0.90	0.59	0.42	0.44	9.0	0.640	0.995	12.0	0.945	0.937	15.0	0.913	0.875	18.0	0.870	0.954	19.0	0.798	0.928
269	84\	OC0684	.SEG	36	1012.6	0.273	2.17	0	0.070	0.217	0.600	0.93	0.76	0.64	0.64	9.0	0.864	1.050	12.0	0.958	0.942	15.0	0.902	0.916	18.0	0.791	0.939	19.0	1.000	1.000
270	84\	OC0784	.SEG	36	1014.1	0.273	2.15	2	0.070	0.213	0.880	0.96	0.73	0.63	0.71	9.0	0.786	0.979	13.0	0.535	0.631	15.0	0.201	0.285	18.0	0.223	0.383	19.0	0.000	1.132
271	84\	OC0964	.SEG	36	1007.5	0.272	2.12	2	0.070	0.197	0.880	0.90	0.78	0.54	0.52	10.8	0.454	0.683	12.0	0.920	0.950	15.0	1.013	1.018	18.0	0.942	1.124	19.0	0.875	0.912
272	84\	OC1184	.SEG	36	1008.6	0.271	2.09	2	0.070	0.194	0.880	0.90	0.63	0.48	0.52	9.0	0.769	0.973	12.0	1.050	0.977	15.0	1.017	0.977	18.0	0.931	1.007	19.0	0.775	1.000
273	84\	OC1284	.SEG	36	1004.2	0.271	2.07	0	0.070	0.190	0.700	0.96	0.73	0.53	0.46	9.0	0.773	1.165	12.0	0.891	0.925	15.0	0.883	0.881	18.0	0.824	1.070	19.0	1.000	1.000
274	84\	OC1384	.SEG	36	999.8	0.271	1.00	1	0.070	0.186	0.946	0.83	0.51	0.37	0.35	9.0	1.148	1.050	12.0	1.080	0.978	15.0	1.063	0.993	18.0	0.940	1.004	19.0	1.000	1.000
275	84\	OC1484	.SEG	36	996.0	0.270	1.00	1	0.070	0.182	0.946	0.86	0.57	0.36	0.33	9.0	1.000	1.020	12.0	1.031	0.972	15.0	1.043	0.991	18.0	0.880	1.007	19.0	1.000	1.000
276	84\	OC1584	.SEG	36	997.7	0.270	2.03	1	0.070	0.179	0.946	0.86	0.59	0.47	0.45	9.0	0.960	1.020	12.0	1.037	0.963	15.0	0.846	0.854	18.0	0.843	1.012	19.0	1.000	1.000
277	84\	OC1684	.SEG	36	1001.6	0.270	2.01	1	0.070	0.178	0.946	0.90	0.78	0.58	0.49	9.0	0.080	0.020	12.0	0.320	0.320	15.0	0.680	0.680	18.0	0.660	0.660	19.0	1.000	1.000

	A	B	C	K	L	M	N	O	P	Q	R	T	U	V	W	X	Y	Z	AA	AB	AC	AD	AE	AF	AG	AH	AI	AJ	AK	AL
278	84\	OC1784	SEG	36	1012.3	0.270	2.00	1	0.070	0.387	0.946	0.93	0.93	0.81	0.76	9.0	0.042	0.131	12.0	0.300	0.300	15.0	0.400	0.400	18.0	0.400	0.400	19.0	1.000	1.000
279	85\	JN2685	SEG	0	997.7	0.323	2.00	0	0.050	0.393	0.600	0.62	0.45	0.38	0.29	9.0	1.134	0.960	12.0	1.020	0.872	15.0	1.000	0.850	18.0	0.845	0.723	19.0	1.000	1.000
280	85\	JN2885	SEG	0	999.1	0.321	2.00	0	0.050	0.413	0.600	0.81	0.43	0.37	0.39	9.0	1.235	1.165	13.0	1.090	0.979	15.0	0.885	0.696	18.0	0.869	0.762	19.0	1.000	1.000
281	85\	AU2985	SEG	0	1006.2	0.291	2.10	0	0.050	0.429	0.600	0.87	0.67	0.57	0.50	10.2	1.063	1.092	12.0	0.901	0.835	15.0	1.027	0.932	18.0	1.021	0.928	19.0	1.000	1.000
282	86\	JL0986	SEG	0	997.4	0.314	2.00	0	0.025	0.442	0.550	0.74	0.46	0.36	0.38	9.0	1.060	1.100	12.0	1.073	0.992	15.0	1.060	0.957	18.0	1.025	0.933	19.0	1.000	1.000
283	86\	JL1386	SEG	0	1000.4	0.312	2.00	1	0.025	0.442	0.946	0.74	0.52	0.40	0.39	9.0	0.987	0.980	14.0	1.036	0.920	15.0	0.795	0.712	18.0	0.479	0.487	19.0	1.000	1.000

	AP	AQ	AR	AS	AT	AU	AV	AW	AX	AY
2										
3	6	11	11	8	9	9	9	9	6	6
4		Sol	Eqt	Sol	Twilight	Sunrise	Noon	Sunset	Tsr	Tnoon
5	Filename	dec, deg	Time,min	radius	HH.ff	HH.ff	HH.ff	HH.ff	HHMM	HHMM
6	MY1677	19.74937	3.59136	1.0118	5.6636	6.1416	13.2140	20.2865	608	1313
7	JN0377	22.42219	1.79961	1.0146	5.5213	6.0160	13.2439	20.4718	601	1310
8	JL1877	20.86188	-6.21731	1.0162	5.7568	6.2413	13.3775	20.5137	614	1323
9	JL2077	20.48822	-6.33549	1.0161	5.7826	6.2648	13.3795	20.4942	616	1323
10	JL2777	19.00362	-6.44864	1.0154	5.8771	6.3510	13.3813	20.4117	621	1323
11	JL2877	18.77005	-6.42292	1.0153	5.8910	6.3637	13.3809	20.3982	622	1323
12	AU0877	15.87888	-5.49724	1.0138	6.0473	6.5062	13.3655	20.2249	630	1322
13	ST1477	3.05980	4.69662	1.0055	6.5534	6.9802	13.1956	19.4110	659	1312
14	ST1977	1.12604	6.47487	1.0042	6.6183	7.0436	13.1660	19.2884	703	1310
15	OC1277	-7.74152	13.66777	0.9976	6.9231	7.3506	13.0461	18.7416	721	1303
16	OC1877	-9.94984	14.94917	0.9959	7.0072	7.4378	13.0247	18.6117	726	1301
17	OC2277	-11.37508	15.59869	0.9948	7.0647	7.4978	13.0139	18.5300	730	1301
18	OC2477	-12.07100	15.85567	0.9943	7.0940	7.5284	13.0096	18.4908	732	1301
19	NO0977	-17.12008	16.06713	0.9902	7.3377	7.7857	13.0061	18.2266	747	1300
20	NO1277	-17.94179	15.71481	0.9895	7.3847	7.8355	13.0120	18.1885	750	1301
21	NO1977	-19.67200	14.40057	0.9880	7.4944	7.9516	13.0339	18.1162	757	1302
22	NO2077	-19.89622	14.15763	0.9878	7.5099	7.9680	13.0379	18.1078	758	1302
23	DE2677	-23.33031	-0.97948	0.9834	7.9420	8.4161	13.2902	18.1643	825	1317
24	JA1078	-21.85512	-7.78802	0.9834	7.9773	8.4440	13.4037	18.3634	827	1324
25	JA2878	-18.01930	-13.09751	0.9850	7.8688	8.3198	13.4922	18.6645	819	1330
26	JA2978	-17.74949	-13.27054	0.9851	7.8582	8.3082	13.4951	18.6819	818	1330
27	FE2778									
28	MR0678	-5.41019	-11.21028	0.9924	7.2267	7.6521	13.4607	19.2693	739	1328
29	MR3178	4.40628	-4.06535	0.9993	6.6329	7.0612	13.3417	19.6221	704	1320
30	JN1678	23.36896	-0.70657	1.0160	5.4992	6.0007	13.2857	20.5706	600	1317
31	JN1878	23.41753	-1.13960	1.0161	5.5031	6.0050	13.2929	20.5808	600	1318
32	JN2178	23.43875	-1.79181	1.0163	5.5126	6.0148	13.3038	20.5930	601	1318
33	JL0178	23.06351	-3.86134	1.0167	5.5726	6.0718	13.3383	20.6047	604	1320
34	JL2178	20.34623	-6.36465	1.0160	5.7921	6.2735	13.3800	20.4865	616	1323
35	JL2278	20.14684	-6.40275	1.0159	5.8053	6.2855	13.3806	20.4757	617	1323
36	JL2478	19.73071	-6.44995	1.0157	5.8322	6.3100	13.3814	20.4528	619	1323
37	JL2778	19.06601	-6.44685	1.0154	5.8733	6.3475	13.3813	20.4152	621	1323
38	JL3078	18.35414	-6.35355	1.0151	5.9152	6.3856	13.3798	20.3739	623	1323
39	AU0578	16.79735	-5.89439	1.0143	6.0004	6.4633	13.3721	20.2809	628	1322
40	AU0678	16.52159	-5.78280	1.0141	6.0147	6.4764	13.3703	20.2642	629	1322
41	AU0878	15.95683	-5.53009	1.0138	6.0433	6.5025	13.3681	20.2296	630	1322
42	AU1178	15.07790	-5.07862	1.0133	6.0862	6.5418	13.3585	20.1753	633	1322
43	AU1478	14.16287	-4.54308	1.0128	6.1290	6.5811	13.3498	20.1182	635	1321
44	AU1578	13.85023	-4.34654	1.0126	6.1432	6.5941	13.3463	20.0986	636	1321
45	AU1678	13.53394	-4.14126	1.0124	6.1573	6.6072	13.3429	20.0787	636	1321
46	AU1778	13.21406	-3.92739	1.0122	6.1715	6.6202	13.3394	20.0585	637	1320
47	AU1878	12.89070	-3.70514	1.0120	6.1856	6.6332	13.3357	20.0381	638	1320
48	AU2178	11.90053	-2.98998	1.0114	6.2276	6.6721	13.3237	19.9754	640	1319
49	AU2378	11.22459	-2.47495	1.0110	6.2554	6.6979	13.3151	19.9324	642	1319
50	AU2478	10.88212	-2.20663	1.0108	6.2693	6.7107	13.3107	19.9106	643	1319
51	AU2578	10.53679	-1.93142	1.0106	6.2831	6.7236	13.3061	19.8886	643	1318
52	ST0878	5.45041	2.49839	1.0072	6.4713	6.9012	13.2323	19.5834	654	1314
53	ST1478	3.16448	4.60484	1.0058	6.5498	6.9767	13.1972	19.4176	659	1312
54	ST1578	2.77951	4.96042	1.0054	6.5628	6.9894	13.1912	19.3931	659	1311
55	ST1878	1.61969	6.02825	1.0045	6.6017	7.0273	13.1734	19.3195	702	1310
56	ST1978	1.23174	6.38345	1.0043	6.6147	7.0400	13.1675	19.2960	702	1310
57	ST2178	0.45439	7.09072	1.0037	6.6407	7.0655	13.1557	19.2459	704	1309
58	OC0378	-4.21055	11.10331	1.0003	6.7976	7.2224	13.0888	18.9553	713	1306
59	OC1278	-7.63948	13.60768	0.9977	6.9192	7.3467	13.0471	18.7475	721	1303
60	OC1378	-8.01309	13.84714	0.9974	6.9330	7.3609	13.0431	18.7253	722	1303
61	OC1778	-9.48790	14.71296	0.9963	6.9890	7.4189	13.0287	18.6385	725	1302
62	OC1878	-9.85117	14.90494	0.9960	7.0032	7.4336	13.0255	18.6174	726	1302
63	OC2078	-10.57042	15.25755	0.9954	7.0318	7.4634	13.0196	18.5758	728	1301
64	OC2178	-10.92619	15.41765	0.9952	7.0462	7.4784	13.0169	18.5555	729	1301
65	OC2278	-11.27924	15.56659	0.9949	7.0606	7.4936	13.0145	18.5354	730	1301
66	OC2578	-12.32101	15.94405	0.9941	7.1046	7.5396	13.0082	18.4768	732	1300
67	OC2978	-13.66530	16.27695	0.9930	7.1642	7.6022	13.0026	18.4030	736	1300
68	NO0278	-14.95191	16.40353	0.9920	7.2249	7.6683	13.0005	18.3347	740	1300
69	NO0778	-16.46810	16.25666	0.9907	7.3021	7.7480	13.0030	18.2579	745	1300
70	MY2179									
71	MY3179									

	AP	AQ	AR	AS	AT	AU	AV	AW	AX	AY
72	JN0179	22.10742	2.21593	1.0142	5.5352	6.0278	13.2370	20.4462	602	1314
73	JN0279	22.23742	2.05880	1.0143	5.5292	6.0227	13.2396	20.4565	601	1314
74	JN1279	23.17509	0.19601	1.0156	5.4974	5.9974	13.2706	20.5438	560	1316
75	JN1379	23.23188	-0.01203	1.0157	5.4970	5.9975	13.2741	20.5507	560	1316
76	JN1479	23.28184	-0.22259	1.0158	5.4971	5.9979	13.2776	20.5573	560	1317
77	JN2079	23.43748	-1.51730	1.0163	5.5081	6.0101	13.2992	20.5883	601	1318
78	JN2779	23.30601	-3.01233	1.0166	5.5420	6.0430	13.3241	20.6052	603	1319
79	JN2979	23.20685	-3.41883	1.0167	5.5555	6.0558	13.3309	20.6060	603	1320
80	JL2179	20.39361	-6.34969	1.0160	5.7888	6.2705	13.3797	20.4889	616	1323
81	JL2979	18.65437	-6.39892	1.0152	5.8977	6.3697	13.3805	20.3914	622	1323
82	JL3079	18.41319	-6.36018	1.0151	5.9117	6.3825	13.3799	20.3773	623	1323
83	JL3179	18.16698	-6.31133	1.0150	5.9258	6.3953	13.3791	20.3629	624	1323
84	AU0179	17.91583	-6.25236	1.0149	5.9400	6.4082	13.3781	20.3480	624	1323
85	AU0279	17.65980	-6.18328	1.0147	5.9541	6.4211	13.3770	20.3328	625	1323
86	AU0479	17.13348	-6.01488	1.0145	5.9826	6.4471	13.3741	20.3012	627	1322
87	AU0579	16.86334	-5.91564	1.0143	5.9969	6.4601	13.3725	20.2849	628	1322
88	AU1379	14.54585	-4.77048	1.0130	6.1112	6.5648	13.3534	20.1421	634	1321
89	AU1579	13.92616	-4.39062	1.0127	6.1397	6.5909	13.3471	20.1033	635	1321
90	AU1679	13.61074	-4.18743	1.0125	6.1538	6.6039	13.3437	20.0835	636	1321
91	ST0779	5.91735	2.07675	1.0075	6.4549	6.8854	13.2393	19.5931	653	1314
92	OC0979	-6.41667	12.77955	0.9986	6.8748	7.3010	13.0609	18.8208	718	1304
93	OC1479	-8.29499	14.02877	0.9972	6.9435	7.3717	13.0401	18.7085	722	1302
94	OC1879	-9.76340	14.86566	0.9961	6.9996	7.4299	13.0261	18.6224	726	1302
95	OC2079	-10.48386	15.22350	0.9955	7.0282	7.4596	13.0202	18.5807	728	1301
96	OC2479	-11.89292	15.80721	0.9944	7.0862	7.5203	13.0104	18.5006	731	1301
97	DE1179	-23.03077	6.62574	0.9846	7.7993	8.2718	13.1635	18.0551	816	1310
98	JN0480	22.56259	1.59830	1.0147	5.5153	6.0109	13.2473	20.4836	601	1315
99	JN1280	23.21904	0.04266	1.0157	5.4970	5.9973	13.2732	20.5490	560	1316
100	JN1480	23.31543	-0.37959	1.0158	5.4974	5.9985	13.2802	20.5619	560	1317
101	JN1580	23.35335	-0.59368	1.0159	5.4984	5.9998	13.2838	20.5678	560	1317
102	JN1780	23.40857	-1.02585	1.0161	5.5019	6.0036	13.2910	20.5784	600	1317
103	JN2080	23.43978	-1.67819	1.0163	5.5108	6.0126	13.3019	20.5911	601	1318
104	JN2180	23.43640	-1.89514	1.0164	5.5144	6.0164	13.3055	20.5948	601	1318
105	JL0880	22.33912	-5.13509	1.0167	5.6424	6.1365	13.3595	20.5825	608	1322
106	JL0980	22.21497	-5.27903	1.0166	5.6530	6.1463	13.3619	20.5775	609	1322
107	JL1580	21.33811	-5.98278	1.0164	5.7222	6.2097	13.3736	20.5375	613	1322
108	JL1680	21.17050	-6.07120	1.0163	5.7345	6.2209	13.3751	20.5292	613	1323
109	JL1780	20.99690	-6.15063	1.0163	5.7470	6.2324	13.3764	20.5205	614	1323
110	JL2580	19.40092	-6.45198	1.0156	5.8527	6.3267	13.3814	20.4341	620	1323
111	JL3080	18.22716	-6.32080	1.0150	5.9223	6.3921	13.3792	20.3664	624	1323
112	AU0180	17.72233	-6.19765	1.0148	5.9506	6.4179	13.3772	20.3365	625	1323
113	AU0880	15.80835	-5.45102	1.0137	6.0506	6.5092	13.3648	20.2203	631	1322
114	AU0980	15.51736	-5.30511	1.0136	6.0649	6.5223	13.3623	20.2024	631	1322
115	AU1180	14.92304	-4.98482	1.0132	6.0935	6.5484	13.3570	20.1655	633	1321
116	AU1280	14.61989	-4.81069	1.0131	6.1077	6.5615	13.3541	20.1466	634	1321
117	AU1380	14.31285	-4.62742	1.0129	6.1220	6.5746	13.3510	20.1274	634	1321
118	AU1480	14.00202	-4.43515	1.0127	6.1362	6.5877	13.3478	20.1080	635	1321
119	AU2680	10.00798	-1.49368	1.0102	6.3037	6.7429	13.2988	19.8547	645	1318
120	AU2780	9.65581	-1.20218	1.0100	6.3174	6.7556	13.2939	19.8322	645	1318
121	JN1581	23.34531	-0.53907	1.0159	5.4981	5.9994	13.2829	20.5664	560	1317
122	JN1681	23.37802	-0.75433	1.0160	5.4994	6.0010	13.2865	20.5720	600	1317
123	JN1781	23.40386	-0.97076	1.0161	5.5013	6.0030	13.2901	20.5772	600	1317
124	JN1881	23.42281	-1.18796	1.0161	5.5036	6.0055	13.2937	20.5819	600	1318
125	JN1981	23.43488	-1.40554	1.0162	5.5064	6.0083	13.2973	20.5863	601	1318
126	JN2181	23.43836	-1.84025	1.0163	5.5134	6.0154	13.3046	20.5938	601	1318
127	JN2281	23.42975	-2.05659	1.0164	5.5176	6.0195	13.3082	20.5968	601	1318
128	JN2381	23.41427	-2.27172	1.0165	5.5222	6.0240	13.3118	20.5995	601	1319
129	JN2481	23.39192	-2.48526	1.0165	5.5273	6.0290	13.3153	20.6017	602	1319
130	JN2581	23.36271	-2.69682	1.0165	5.5328	6.0343	13.3188	20.6034	602	1319
131	JN2981	23.17758	-3.51551	1.0167	5.5591	6.0591	13.3325	20.6058	604	1320
132	JL0681	22.59450	-4.78567	1.0167	5.6195	6.1154	13.3537	20.5919	607	1321
133	JL0881	22.36864	-5.09723	1.0167	5.6398	6.1341	13.3589	20.5836	608	1322
134	JL0981	22.24603	-5.24289	1.0166	5.6504	6.1438	13.3613	20.5787	609	1322
135	JL1581	21.37817	-5.95824	1.0164	5.7192	6.2069	13.3732	20.5395	612	1322
136	JL1881	20.86175	-6.20341	1.0162	5.7565	6.2411	13.3773	20.5135	614	1323
137	JL2081	20.48805	-6.32160	1.0161	5.7823	6.2646	13.3793	20.4939	616	1323
138	JL2181	20.29257	-6.36664	1.0160	5.7955	6.2768	13.3800	20.4834	617	1323
139	JL2281	20.09143	-6.40214	1.0159	5.8088	6.2887	13.3806	20.4725	617	1323
140	JL2381	19.88470	-6.42797	1.0158	5.8222	6.3009	13.3810	20.4612	618	1323
141	JL3181	18.03843	-6.27719	1.0149	5.9330	6.4018	13.3785	20.3552	624	1323

	AP	AQ	AR	AS	AT	AU	AV	AW	AX	AY
142	AU0581	16.72255	-5.85557	1.0142	6.0042	6.4667	13.3715	20.2762	628	1322
143	AU1381	14.38773	-4.67083	1.0129	6.1185	6.5714	13.3517	20.1321	634	1321
144	AU1481	14.07780	-4.48072	1.0128	6.1327	6.5845	13.3486	20.1127	635	1321
145	AU2681	10.09296	-1.58128	1.0103	6.3004	6.7397	13.2999	19.8601	644	1318
146	AU2781	9.74142	-1.27125	1.0101	6.3141	6.7525	13.2951	19.8377	645	1318
147	AU3181	8.31033	-0.05304	1.0091	6.3683	6.8034	13.2748	19.7461	648	1316
148	ST0281	7.58108	0.58709	1.0087	6.3952	6.8288	13.2641	19.6995	650	1316
149	ST0381	7.21334	0.91393	1.0084	6.4085	6.8414	13.2587	19.6759	650	1316
150	ST0981	4.96904	2.95092	1.0069	6.4879	6.9170	13.2247	19.5324	655	1313
151	ST1081	4.58951	3.30021	1.0066	6.5010	6.9296	13.2189	19.5082	656	1313
152	ST1281	3.82660	4.00434	1.0061	6.5271	6.9548	13.2072	19.4595	657	1312
153	ST1481	3.05913	4.71370	1.0056	6.5532	6.9800	13.1953	19.4107	659	1312
154	ST1781	1.90100	5.78179	1.0047	6.5921	7.0179	13.1775	19.3371	701	1311
155	ST2081	0.73679	6.84674	1.0039	6.6310	7.0561	13.1598	19.2635	703	1310
156	ST2481	-0.82060	8.24598	1.0028	6.6830	7.1074	13.1365	19.1655	706	1308
157	ST2881	-2.37827	9.60158	1.0017	6.7351	7.1594	13.1139	19.0683	710	1307
158	ST2981	-2.76701	9.93110	1.0014	6.7483	7.1726	13.1084	19.0442	710	1307
159	OC0181	-3.54299	10.57649	1.0008	6.7746	7.1991	13.0976	18.9961	712	1306
160	OC1481	-8.48644	14.15481	0.9971	6.9505	7.3790	13.0380	18.6970	723	1302
161	NO1381	-18.20600	15.59710	0.9893	7.4000	7.8517	13.0139	18.1762	751	1301
162	NO1581	-18.71726	15.26372	0.9888	7.4314	7.8849	13.0195	18.1541	753	1301
163	NO2281	-20.32706	13.66145	0.9874	7.5403	8.0003	13.0462	18.0922	800	1303
164	JN0682	22.72106	1.32895	1.0149	5.5092	6.0059	13.2518	20.4976	600	1315
165	JN0882	22.90521	0.95096	1.0152	5.5031	6.0012	13.2581	20.5149	600	1315
166	JN0982	22.98724	0.75545	1.0153	5.5008	5.9995	13.2613	20.5231	560	1316
167	JN1482	23.29579	-0.27313	1.0158	5.4970	5.9979	13.2785	20.5590	560	1317
168	JN1582	23.33702	-0.48645	1.0159	5.4978	5.9990	13.2820	20.5650	560	1317
169	JN1682	23.37140	-0.70142	1.0160	5.4990	6.0005	13.2856	20.5707	600	1317
170	JN2582	23.37105	-2.64548	1.0165	5.5314	6.0329	13.3180	20.6031	602	1319
171	JN2782	23.29541	-3.06247	1.0166	5.5435	6.0444	13.3249	20.6054	603	1319
172	JN2982	23.19250	-3.46729	1.0167	5.5573	6.0574	13.3317	20.6059	603	1320
173	JN3082	23.13088	-3.66420	1.0167	5.5647	6.0645	13.3350	20.6055	604	1320
174	JL0182	23.06252	-3.85696	1.0167	5.5726	6.0718	13.3382	20.6046	604	1320
175	JL0382	22.90569	-4.22860	1.0167	5.5894	6.0875	13.3444	20.6013	605	1321
176	JL0682	22.62068	-4.74623	1.0167	5.6171	6.1132	13.3530	20.5928	607	1321
177	JL0882	22.39792	-5.06099	1.0167	5.6372	6.1318	13.3582	20.5847	608	1321
178	JL1582	21.41801	-5.93523	1.0164	5.7162	6.2042	13.3728	20.5414	612	1322
179	JL1782	21.08257	-6.11182	1.0163	5.7408	6.2267	13.3758	20.5248	614	1323
180	JL1982	20.72336	-6.25282	1.0162	5.7662	6.2499	13.3781	20.5063	615	1323
181	JL2682	19.28661	-6.44866	1.0155	5.8597	6.3351	13.3814	20.4276	620	1323
182	JL2782	19.05951	-6.43738	1.0154	5.8735	6.3477	13.3812	20.4147	621	1323
183	JL2882	18.82717	-6.41605	1.0153	5.8874	6.3603	13.3808	20.4013	622	1323
184	AU0282	17.58962	-6.15793	1.0147	5.9579	6.4246	13.3785	20.3285	625	1323
185	AU0382	17.32750	-6.07599	1.0146	5.9721	6.4375	13.3752	20.3128	626	1323
186	AU0682	16.51335	-5.77010	1.0141	6.0150	6.4766	13.3701	20.2635	629	1322
187	AU0782	16.23299	-5.64830	1.0140	6.0293	6.4897	13.3680	20.2484	629	1322
188	AU2082	12.22366	-3.21889	1.0116	6.2139	6.6594	13.3275	19.9957	640	1320
189	AU2282	11.55367	-2.71818	1.0112	6.2418	6.6852	13.3192	19.9532	641	1319
190	AU2382	11.21406	-2.45668	1.0110	6.2557	6.6981	13.3148	19.9316	642	1319
191	AU2482	10.87150	-2.18805	1.0108	6.2695	6.7110	13.3104	19.9098	643	1319
192	AU2782	9.82693	-1.34183	1.0101	6.3108	6.7494	13.2963	19.8431	645	1318
193	ST0382	7.30266	0.83485	1.0085	6.4053	6.8383	13.2600	19.6816	650	1316
194	ST0482	6.93343	1.16488	1.0082	6.4186	6.8510	13.2545	19.6580	651	1315
195	ST0582	6.56231	1.49880	1.0080	6.4319	6.8636	13.2489	19.6343	652	1315
196	ST0682	6.18939	1.83632	1.0077	6.4451	6.8762	13.2433	19.6104	653	1315
197	ST0782	5.81477	2.17712	1.0075	6.4584	6.8888	13.2376	19.5865	653	1314
198	ST0882	5.43853	2.52091	1.0072	6.4715	6.9014	13.2319	19.5624	654	1314
199	ST1282	3.91925	3.91958	1.0062	6.5239	6.9517	13.2086	19.4654	657	1313
200	ST1382	3.53630	4.27353	1.0059	6.5370	6.9643	13.2027	19.4410	658	1312
201	ST1482	3.15228	4.62653	1.0056	6.5500	6.9769	13.1968	19.4166	659	1312
202	ST1682	2.38140	5.34041	1.0051	6.5760	7.0022	13.1849	19.3678	700	1311
203	ST1782	1.99473	5.69682	1.0048	6.5890	7.0148	13.1790	19.3431	701	1311
204	ST1882	1.60736	6.05256	1.0045	6.6019	7.0275	13.1730	19.3185	702	1310
205	ST2482	-0.72631	8.16368	1.0029	6.6798	7.1042	13.1378	19.1714	706	1308
206	ST2982	-2.67301	9.85332	1.0014	6.7451	7.1694	13.1097	19.0500	710	1307
207	OC0182	-3.44928	10.50111	1.0009	6.7714	7.1959	13.0989	19.0019	712	1306
208	OC0382	-4.22299	11.12877	1.0003	6.7978	7.2226	13.0884	18.9542	713	1305
209	OC0682	-5.37704	12.02666	0.9994	6.8378	7.2633	13.0735	18.8837	716	1304
210	OC0882	-6.14088	12.59232	0.9989	6.8648	7.2908	13.0640	18.8373	717	1304
211	OC2782	-13.01108	16.15888	0.9935	7.1344	7.5709	13.0046	18.4383	734	1300

	AP	AQ	AR	AS	AT	AU	AV	AW	AX	AY
212	OC2882	-13.34543	16.23576	0.9933	7.1493	7.5866	13.0033	18.4200	736	1300
213	NO1182	-17.60804	15.90741	0.9898	7.3648	7.8144	13.0088	18.2032	749	1301
214	NO1582	-18.65698	15.31092	0.9889	7.4275	7.8809	13.0187	18.1566	753	1301
215	DE0782	-22.68373	8.32085	0.9850	7.7525	8.2233	13.1352	18.0471	813	1308
216	JN1483	23.28551	-0.22327	1.0158	5.4969	5.9977	13.2776	20.5575	560	1317
217	JN2783	23.30672	-3.01461	1.0166	5.5420	6.0430	13.3241	20.6053	603	1319
218	JL0283	23.00691	-4.00227	1.0167	5.5788	6.0776	13.3406	20.6036	605	1320
219	JL0883	22.42690	-5.02664	1.0167	5.6347	6.1295	13.3577	20.5859	608	1321
220	JL0983	22.30739	-5.17570	1.0167	5.6452	6.1391	13.3602	20.5812	608	1322
221	JL1583	21.45756	-5.91400	1.0164	5.7133	6.2016	13.3725	20.5434	612	1322
222	JL1783	21.12501	-6.09482	1.0163	5.7378	6.2240	13.3755	20.5270	613	1323
223	JL2183	20.38894	-6.34883	1.0160	5.7891	6.2708	13.3797	20.4887	616	1323
224	JL2483	19.77686	-6.44033	1.0158	5.8291	6.3072	13.3812	20.4552	618	1323
225	JL2683	19.34128	-6.45243	1.0156	5.8564	6.3321	13.3814	20.4308	620	1323
226	JL2783	19.11545	-6.44356	1.0155	5.8702	6.3447	13.3813	20.4179	621	1323
227	JL2983	18.64810	-6.39570	1.0152	5.8980	6.3700	13.3805	20.3910	622	1323
228	JL3183	18.16033	-6.30748	1.0150	5.9262	6.3956	13.3790	20.3624	624	1323
229	AU0183	17.90899	-6.24816	1.0149	5.9403	6.4085	13.3780	20.3476	625	1323
230	AU0483	17.12611	-6.00986	1.0145	5.9830	6.4474	13.3741	20.3007	627	1322
231	AU0683	16.58095	-5.80054	1.0142	6.0115	6.4735	13.3706	20.2677	628	1322
232	AU1183	15.14157	-5.10626	1.0134	6.0830	6.5389	13.3590	20.1791	632	1322
233	AU1383	14.53702	-4.76210	1.0130	6.1116	6.5651	13.3533	20.1415	634	1321
234	AU1483	14.22891	-4.57630	1.0128	6.1258	6.5781	13.3502	20.1222	635	1321
235	AU1883	12.95966	-3.74524	1.0121	6.1824	6.6303	13.3383	20.0424	638	1320
236	AU1983	12.63357	-3.51640	1.0119	6.1965	6.6433	13.3325	20.0217	639	1320
237	AU2283	11.63567	-2.78242	1.0113	6.2385	6.6821	13.3203	19.9584	641	1319
238	AU2683	10.26259	-1.70118	1.0104	6.2938	6.7335	13.3023	19.8710	644	1318
239	ST1283	4.01184	3.83290	1.0082	6.5208	6.9487	13.2100	19.4713	657	1313
240	ST2383	-0.24244	7.73136	1.0032	6.6637	7.0883	13.1450	19.2018	705	1309
241	ST2583	-1.02167	8.42454	1.0027	6.6897	7.1140	13.1335	19.1529	707	1308
242	ST2783	-1.80078	9.10610	1.0021	6.7157	7.1400	13.1221	19.1042	708	1307
243	OC0483	-4.51548	11.36069	1.0001	6.8079	7.2328	13.0848	18.9363	714	1305
244	OC0683	-5.28438	11.95631	0.9995	6.8346	7.2600	13.0746	18.8893	716	1304
245	OC0783	-5.66722	12.24435	0.9992	6.8480	7.2737	13.0698	18.8660	716	1304
246	OC1783	-9.41176	14.69030	0.9964	6.9857	7.4155	13.0291	18.6426	725	1302
247	JN2784	23.27284	-3.17462	1.0166	5.5469	6.0477	13.3268	20.6059	603	1320
248	AU0484	16.92202	-5.93967	1.0144	5.9938	6.4573	13.3729	20.2885	627	1322
249	AU0584	16.64825	-5.83252	1.0142	6.0081	6.4704	13.3711	20.2718	628	1322
250	AU0684	16.37001	-5.71545	1.0141	6.0224	6.4834	13.3692	20.2549	629	1322
251	AU0784	16.08738	-5.58855	1.0139	6.0367	6.4965	13.3670	20.2376	630	1322
252	AU0884	15.80044	-5.45191	1.0137	6.0510	6.5096	13.3648	20.2199	631	1322
253	AU0984	15.50928	-5.30563	1.0136	6.0654	6.5227	13.3623	20.2020	631	1322
254	AU2284	11.37930	-2.59022	1.0111	6.2491	6.6919	13.3171	19.9422	642	1319
255	AU2384	11.03814	-2.32494	1.0109	6.2629	6.7048	13.3126	19.9205	642	1319
256	AU2584	10.34716	-1.77381	1.0105	6.2906	6.7305	13.3035	19.8764	644	1318
257	AU2784	9.64524	-1.19622	1.0100	6.3179	6.7561	13.2938	19.8316	645	1318
258	ST0184	7.84726	0.34762	1.0088	6.3855	6.8196	13.2681	19.7166	649	1316
259	ST0284	7.48098	0.67140	1.0086	6.3989	6.8323	13.2627	19.6931	650	1316
260	ST0384	7.11267	0.99952	1.0084	6.4122	6.8449	13.2572	19.6698	651	1315
261	ST0884	5.24401	2.69468	1.0071	6.4784	6.9079	13.2290	19.5500	654	1314
262	ST1684	2.18207	5.52029	1.0050	6.5828	7.0088	13.1819	19.3550	701	1311
263	ST1784	1.79501	5.87650	1.0047	6.5957	7.0214	13.1760	19.3305	701	1311
264	ST1984	1.01903	6.58725	1.0041	6.6217	7.0468	13.1641	19.2814	703	1310
265	ST2184	0.24119	7.29355	1.0036	6.6476	7.0724	13.1523	19.2323	704	1309
266	ST2584	-1.31703	8.68185	1.0024	6.6996	7.1239	13.1292	19.1345	707	1308
267	OC0484	-4.80744	11.58729	0.9999	6.8180	7.2431	13.0808	18.9185	715	1305
268	OC0584	-5.19165	11.88355	0.9998	6.8314	7.2567	13.0758	18.8950	715	1305
269	OC0684	-5.57477	12.17327	0.9993	6.8448	7.2704	13.0710	18.8717	716	1304
270	OC0784	-5.95670	12.45613	0.9990	6.8583	7.2841	13.0663	18.8485	717	1304
271	OC0984	-6.71658	12.99093	0.9984	6.8854	7.3119	13.0572	18.8026	719	1303
272	OC1184	-7.47047	13.51240	0.9979	6.9128	7.3400	13.0487	18.7574	720	1303
273	OC1284	-7.84492	13.75610	0.9976	6.9265	7.3542	13.0446	18.7350	721	1303
274	OC1384	-8.21755	13.99103	0.9973	6.9404	7.3685	13.0407	18.7129	722	1302
275	OC1484	-8.58827	14.21689	0.9970	6.9543	7.3829	13.0370	18.6910	723	1302
276	OC1584	-8.95697	14.43338	0.9967	6.9683	7.3974	13.0333	18.6693	724	1302
277	OC1684	-9.32354	14.64020	0.9964	6.9824	7.4120	13.0299	18.6478	725	1302
278	OC1784	-9.68788	14.83707	0.9962	6.9965	7.4267	13.0266	18.6266	726	1302
279	JN2685	23.32797	-2.92513	1.0166	5.5390	6.0402	13.3227	20.6051	602	1319
280	JN2885	23.23492	-3.33490	1.0166	5.5522	6.0527	13.3295	20.6063	603	1320
281	AU2985	9.01992	-0.67427	1.0096	6.3419	6.7786	13.2851	19.7917	647	1317

	AP	AQ	AR	AS	AT	AU	AV	AW	AX	AY
282	JL0986	22.27492	-5.23489	1.0167	5.6483	6.1420	13.3611	20.5803	608	1321
283	JL1386	21.72632	-5.75028	1.0165	5.6930	6.1831	13.3697	20.5564	610	1322
284										
285										
286										
287										
288										
289										

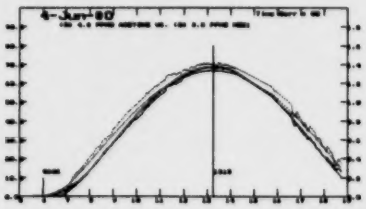
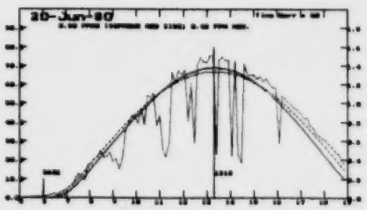
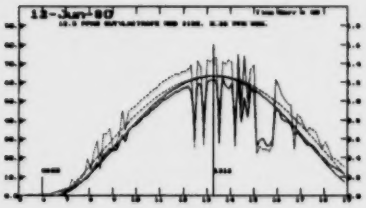
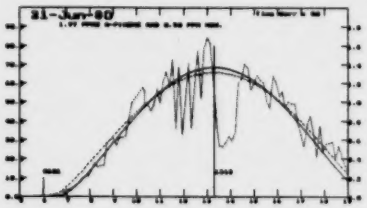
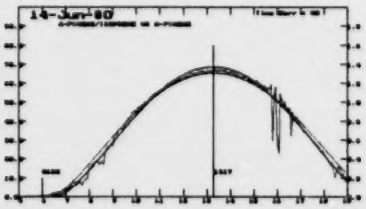
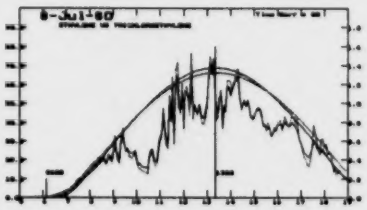
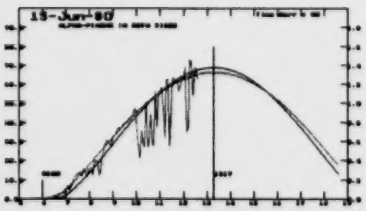
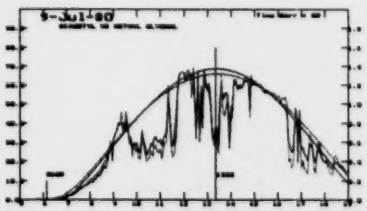
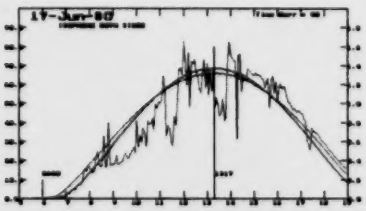
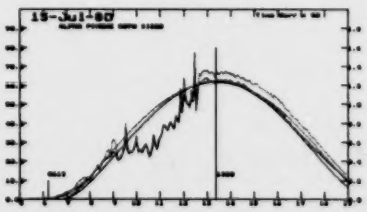
B

Model Plots

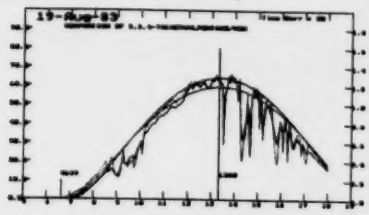
This Appendix contains three sets of TSR and UV plots:

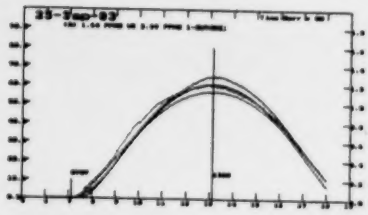
1. Examples of typical "fits" of the clear sky irradiance model to both TSR and UV site radiometers using the data from columns K to W of the SOLCLIM data base given in Appendix A. Cloud effects are not accounted for in this mode. Data from three different years are presented.
2. The complete set of Cloudy Model predictions and TSR and UV site radiometer profiles for every day in the UNC SegFile data base. These plots can be used to:
   a) assess the quality of a particular day relative to other days in the data set;
   b) assess the extent of cloud effects on a given day;
   c) assess the quality of the broadband model predictions of irradiance for the set and for given days;
   d) judge the adequacy of the time intervals that were chosen for a given day for the cloud corrections;
   e) decide whether to model a given day.
3. Examples of NO_2 and HCHOr photolysis rates computed for inside the chamber also plotted with the tower UV radiometer readings. These plots illustrate the high degree of correlation between the horizontal irradiance from the UV radiometer and the chamber rates. This is due to the large horizontal floor reflection component in the chamber. In addition, these plots show the late morning "peaking" of the rates for the fall days, when the sun is directly normal to the Field side chamber face and maximum transmission occurs.

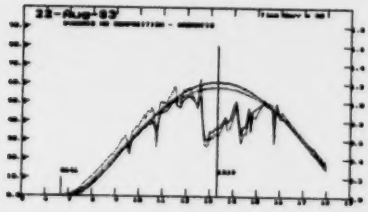
Sample Clear Model Plots

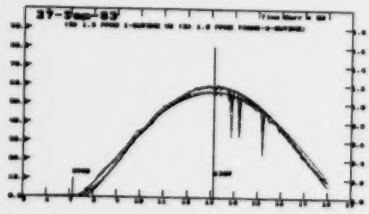











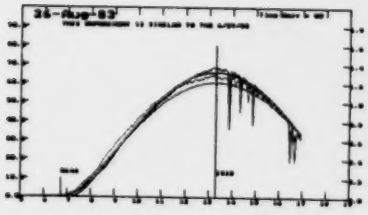
Clear Model

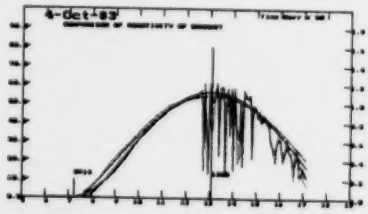


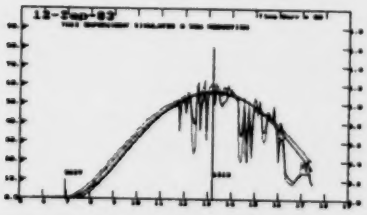


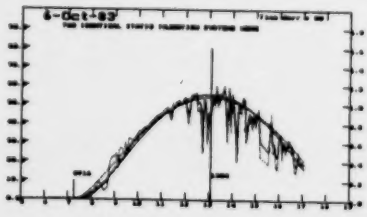


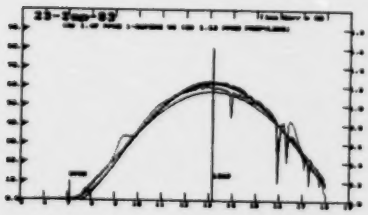


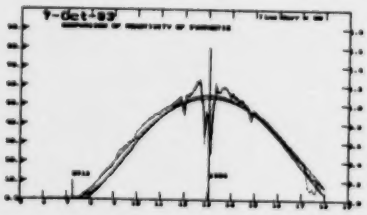




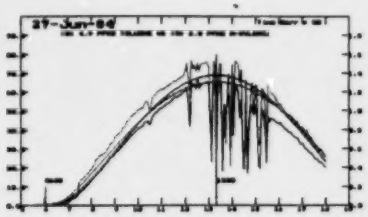


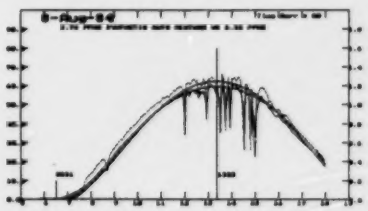


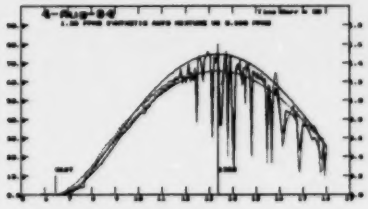


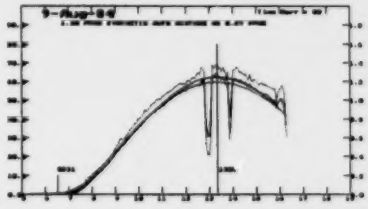


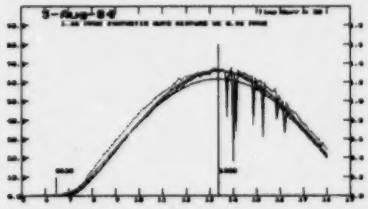
Clear Model

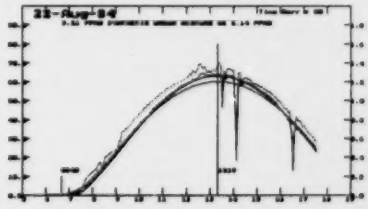


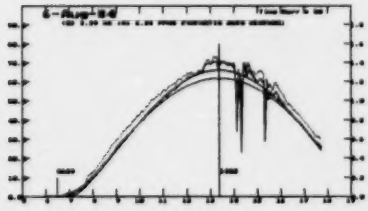


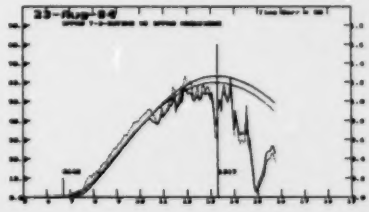


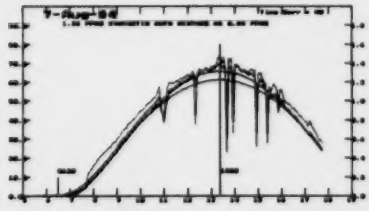


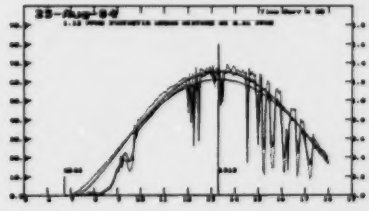




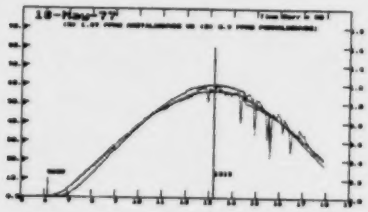


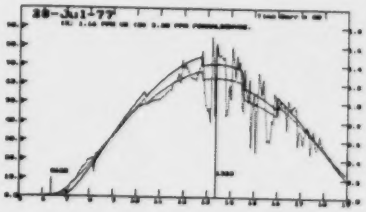


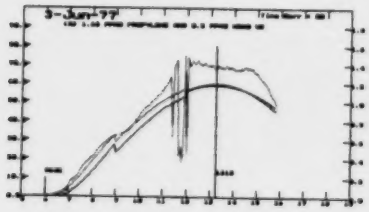


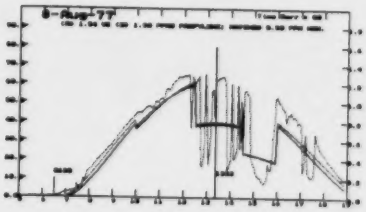


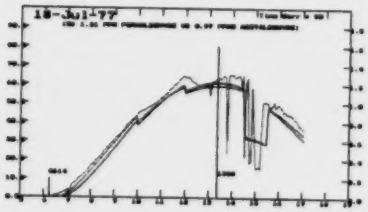
Cloudy Model Plots

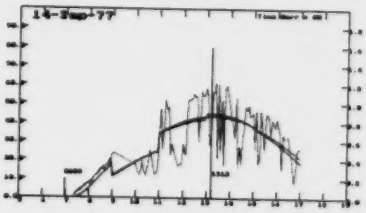


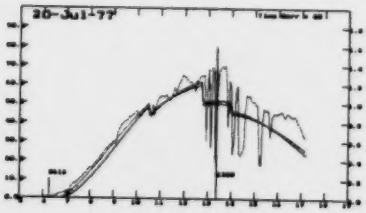


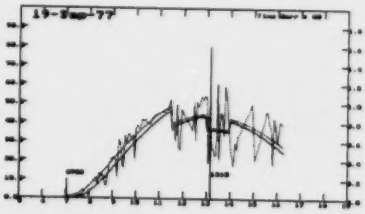


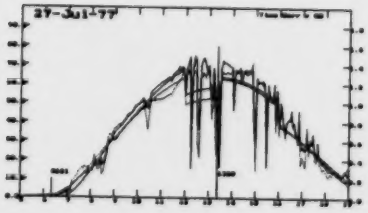


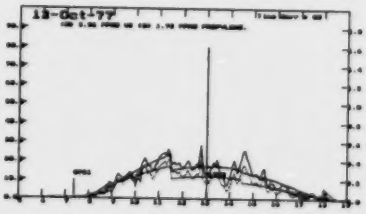




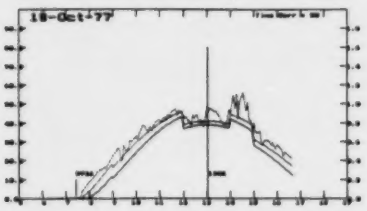


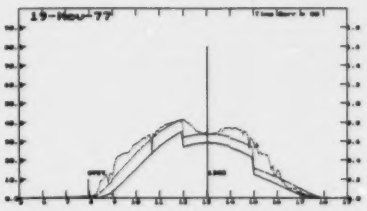


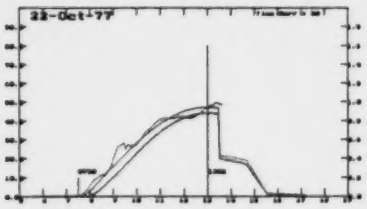


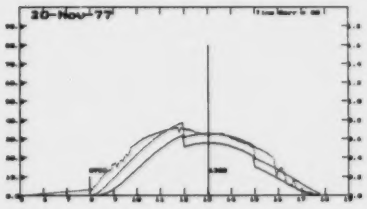


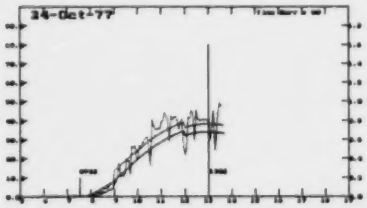
Cloudy Model

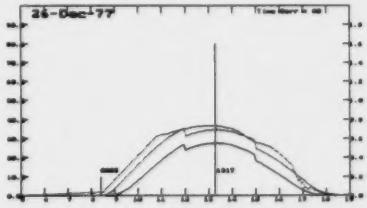


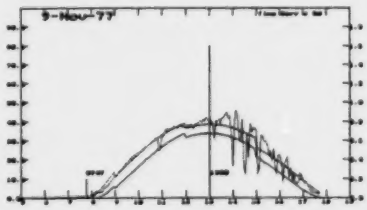


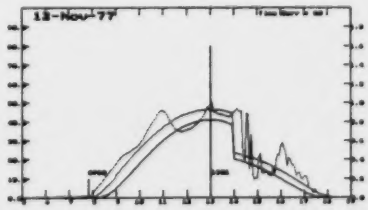




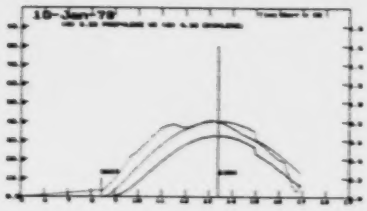


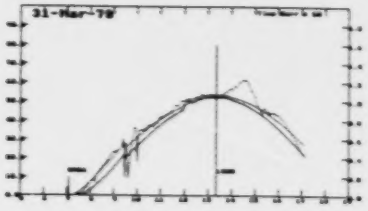


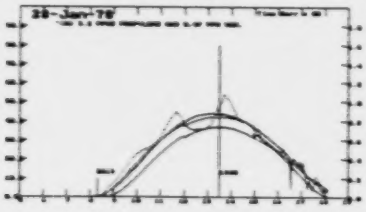


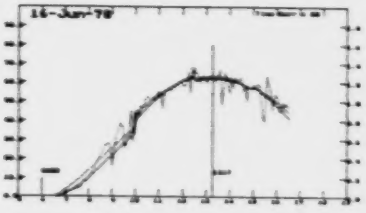


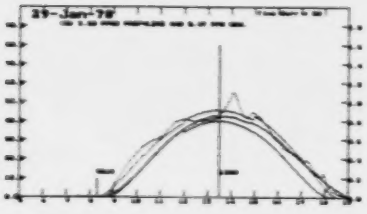
Cloudy Model

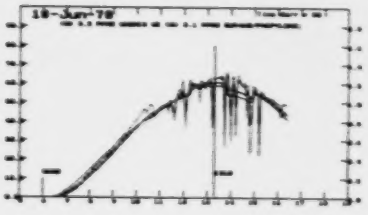


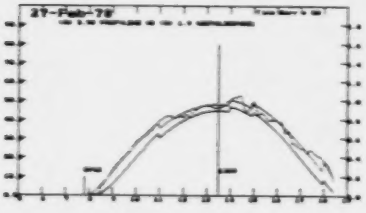


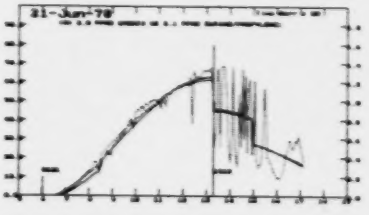


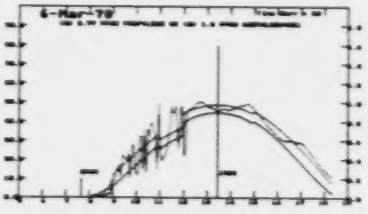


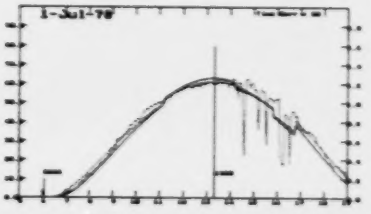




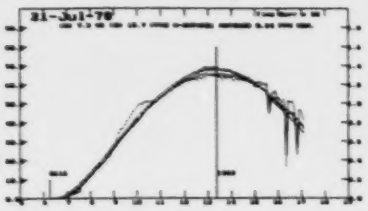


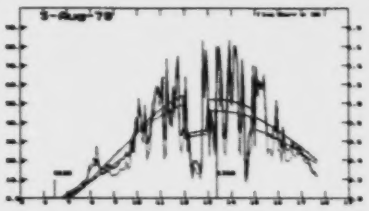


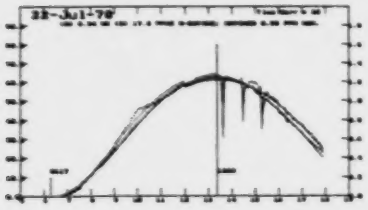


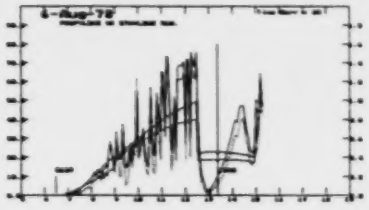


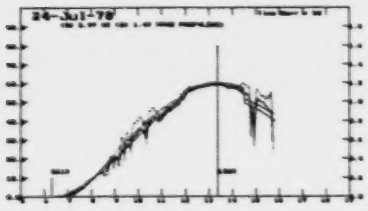
Cloudy Model

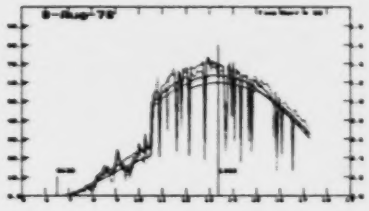


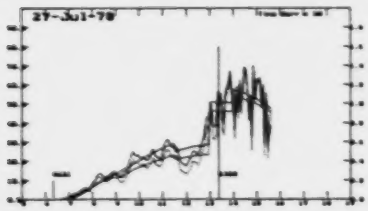


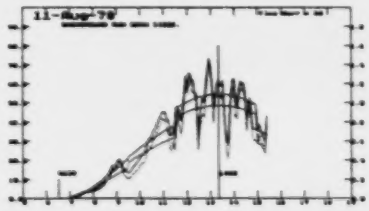


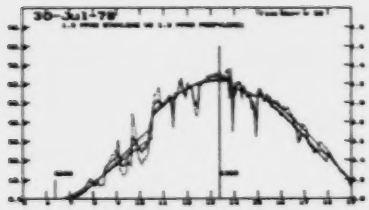


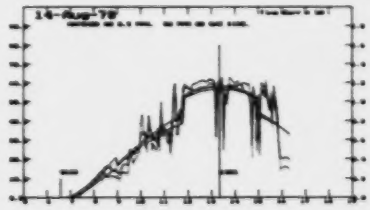




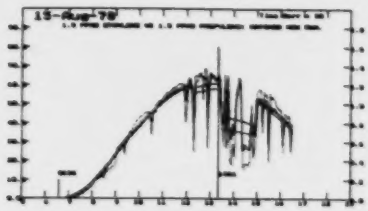


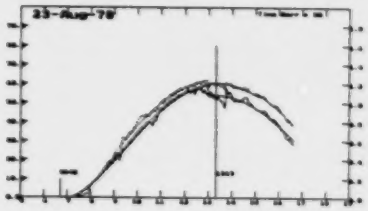


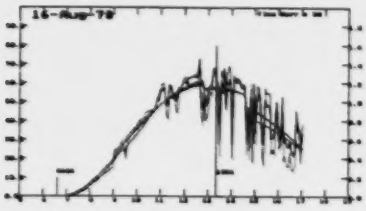


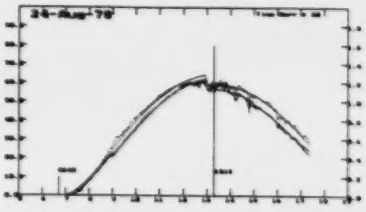


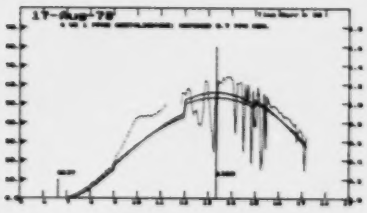
Cloudy Model

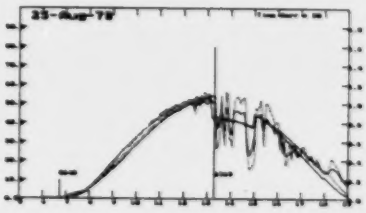


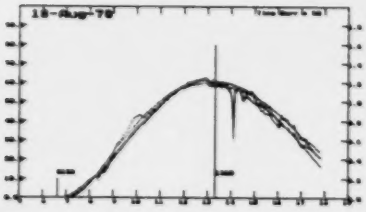


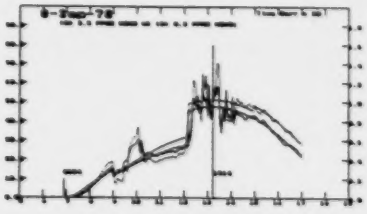


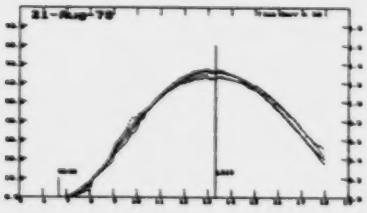


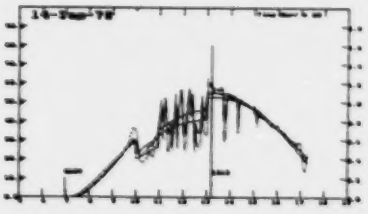




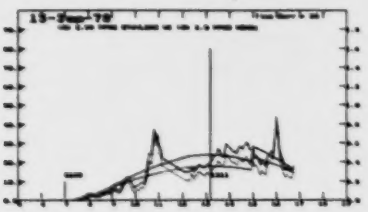


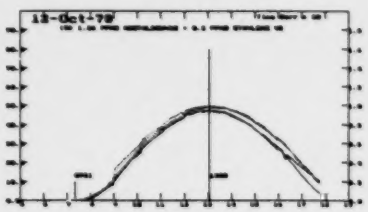


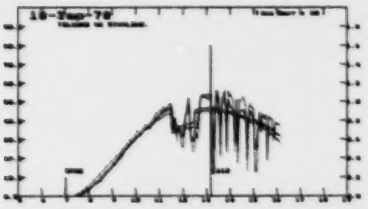


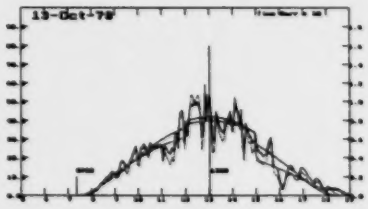


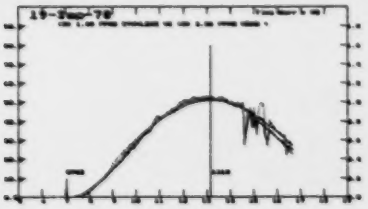
Cloudy Model

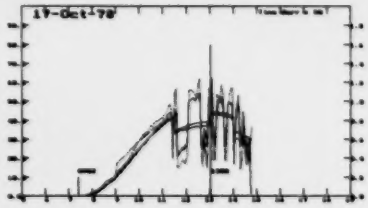


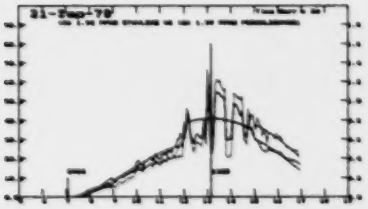


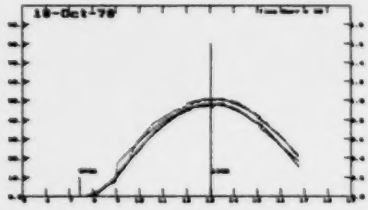


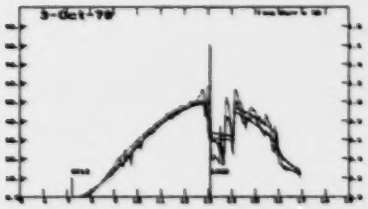


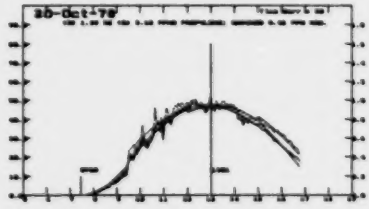




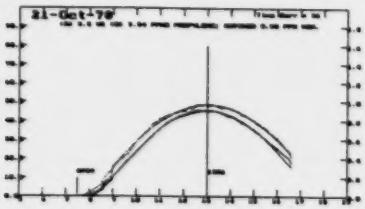


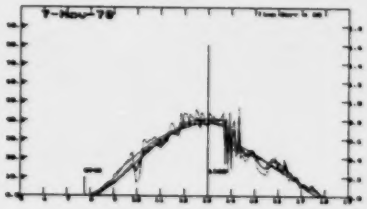


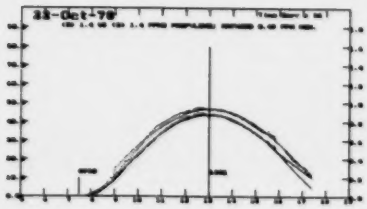


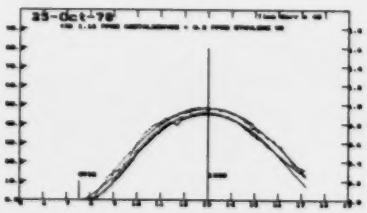


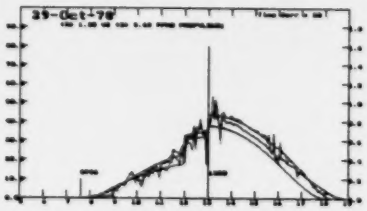
Cloudy Model

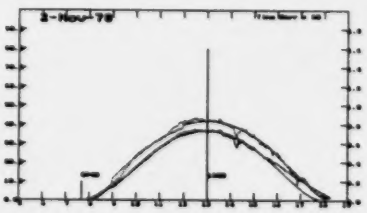




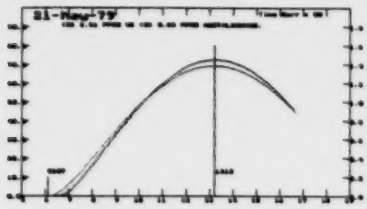


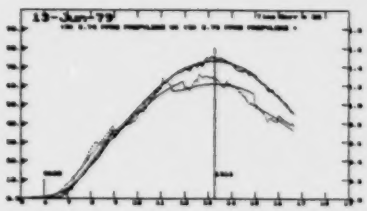


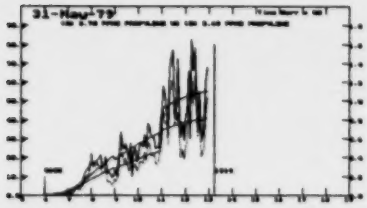


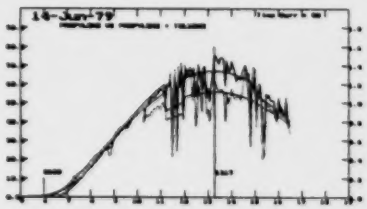


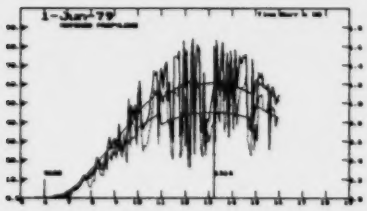
Cloudy Model

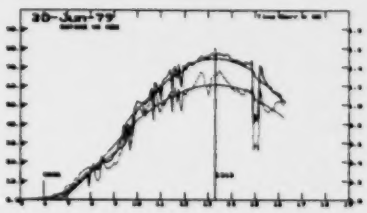


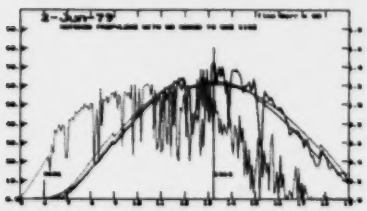


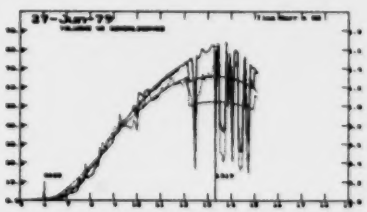


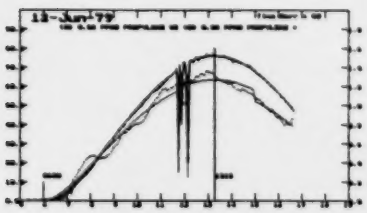


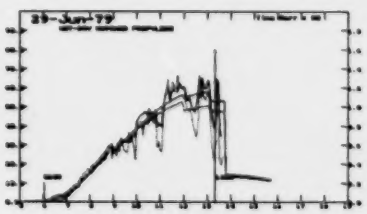




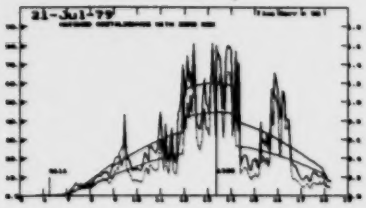


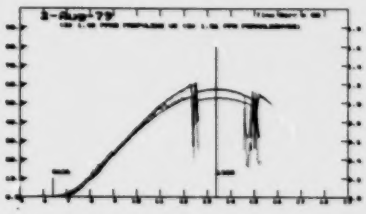


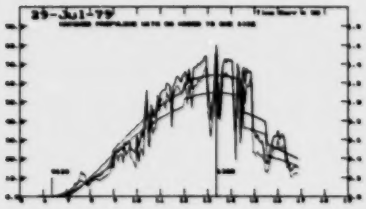


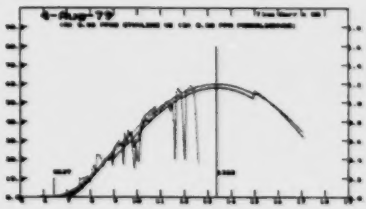


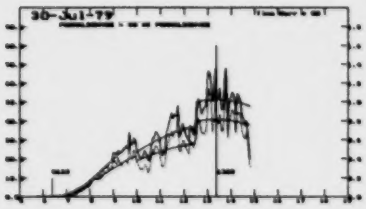
Cloudy Model

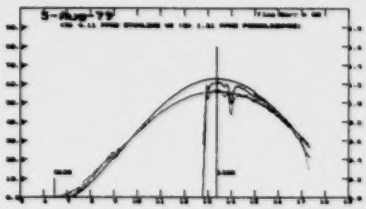


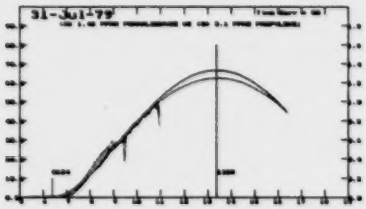


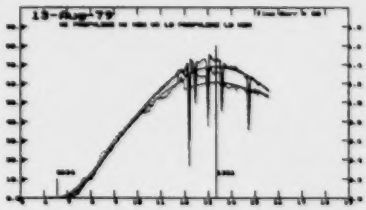


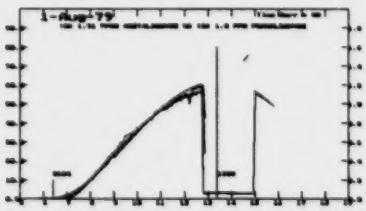


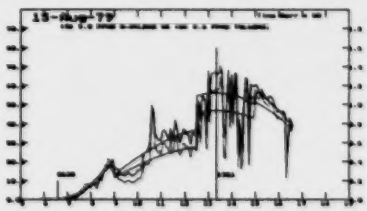




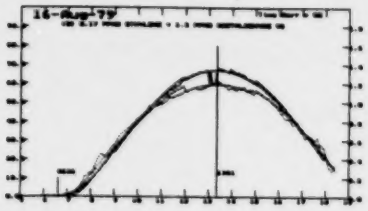


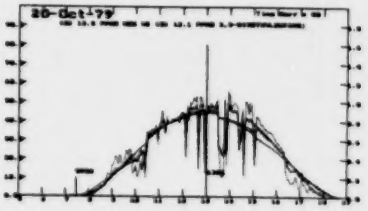


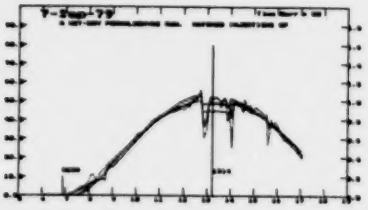


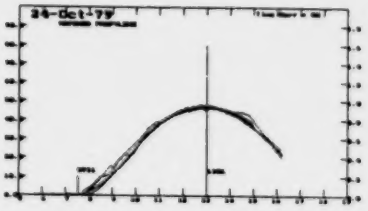


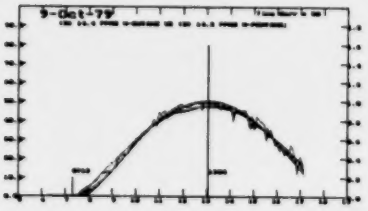
Cloudy Model

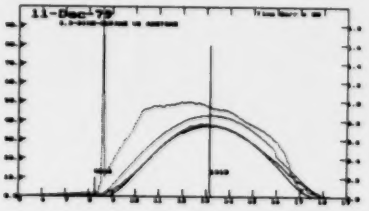


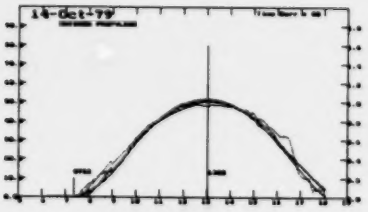


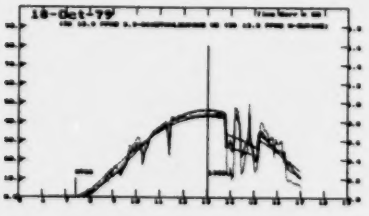




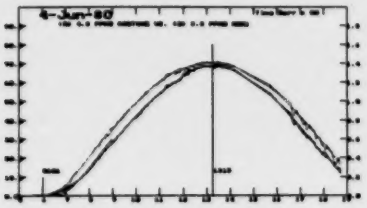


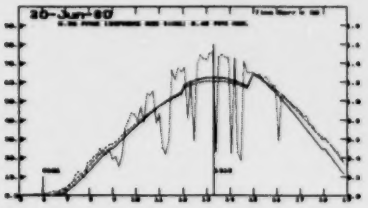


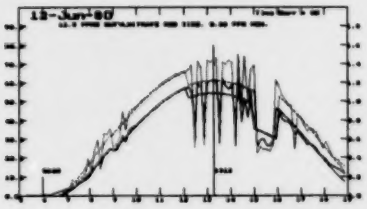


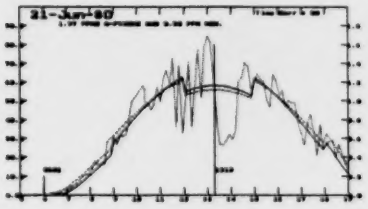


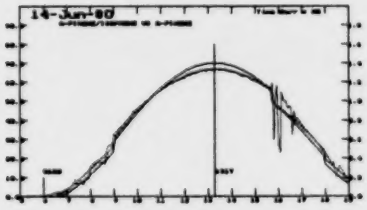
Cloudy Model

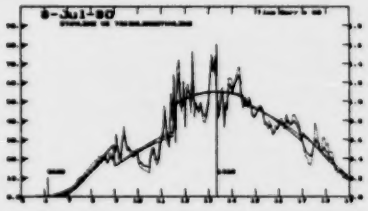


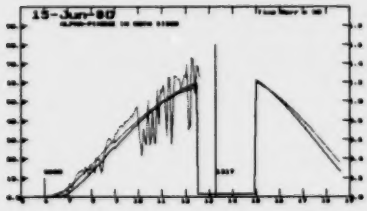


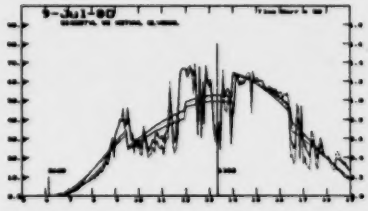


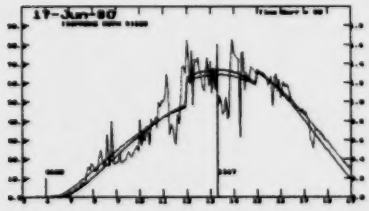


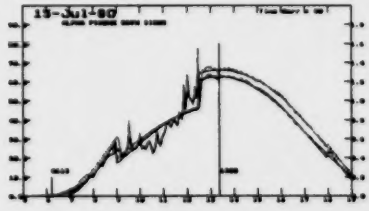




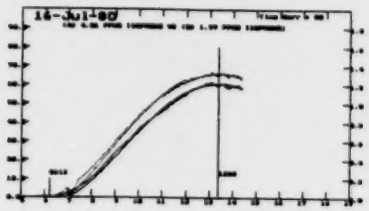


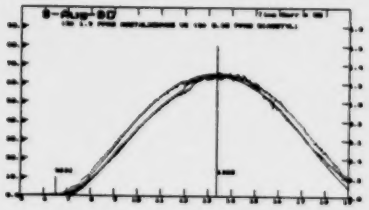


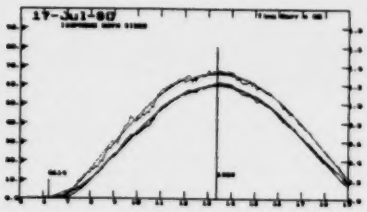


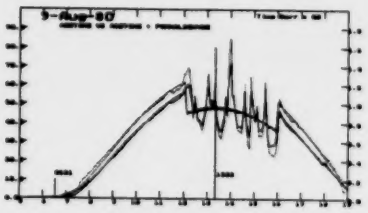


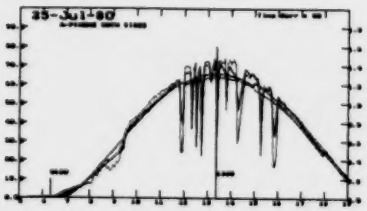
Cloudy Model

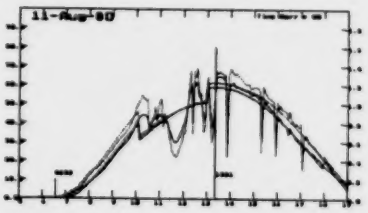


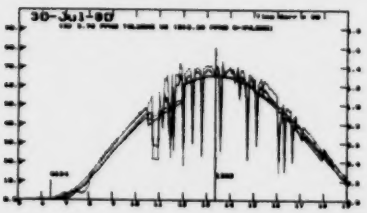


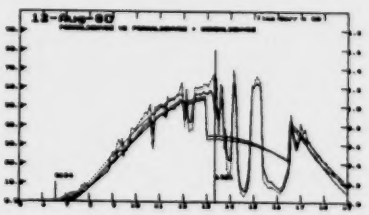


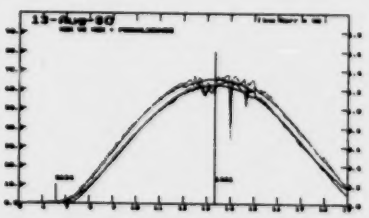


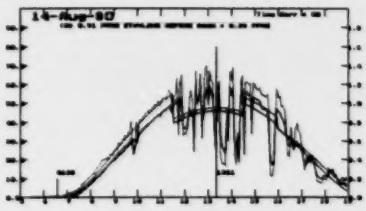


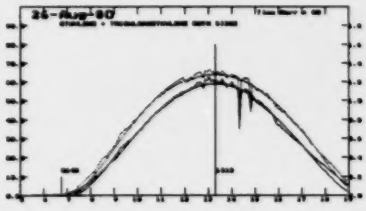


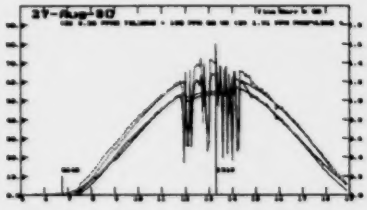




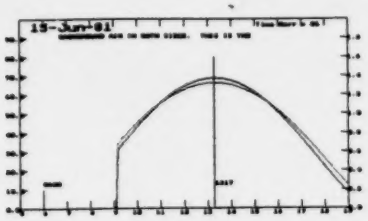


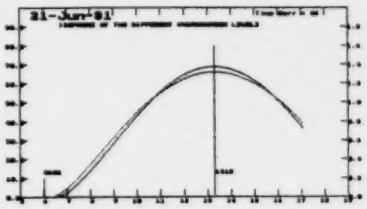


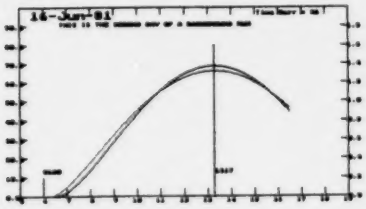


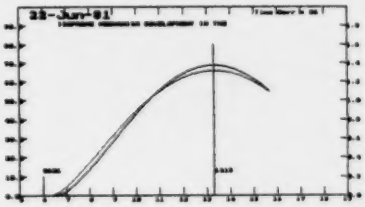


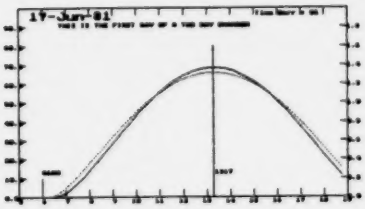
Cloudy Model

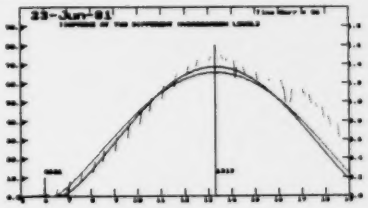


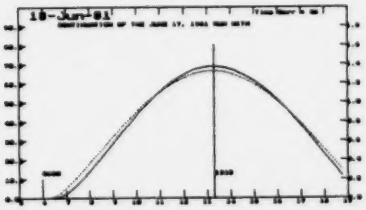


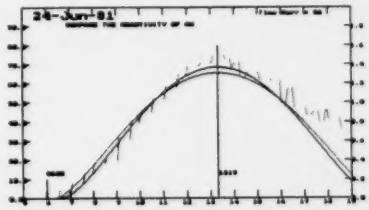


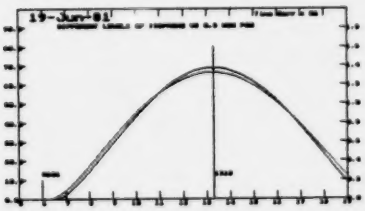


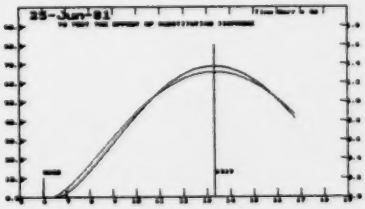




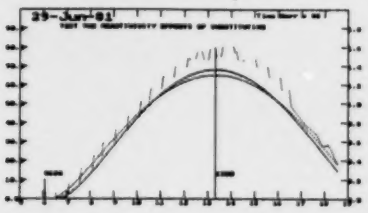


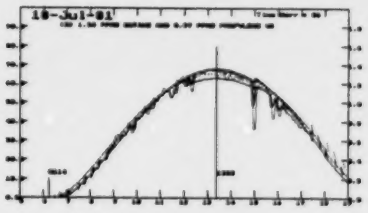


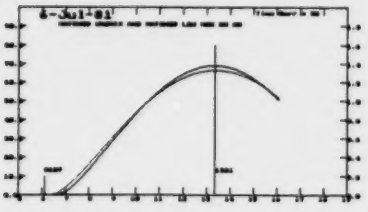


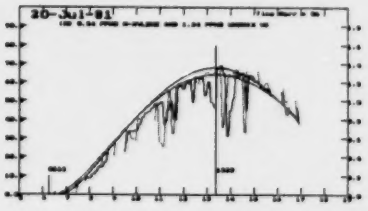


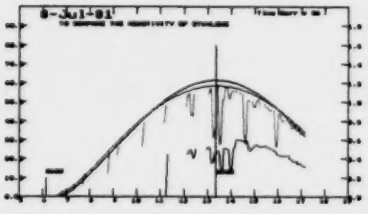
Cloudy Model

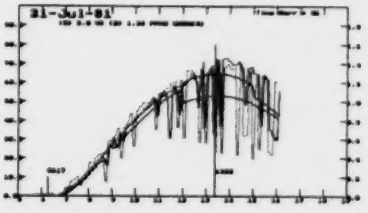


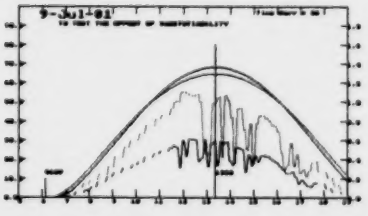


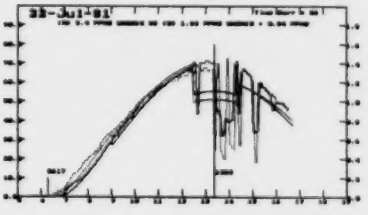


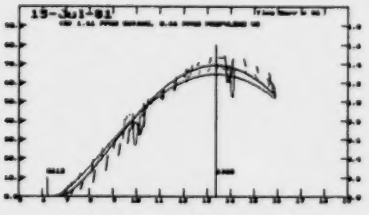


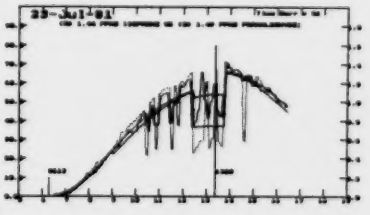




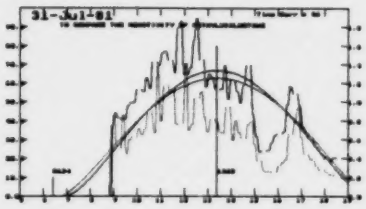


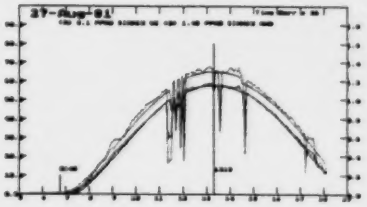


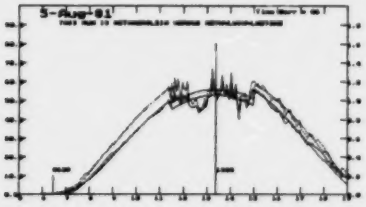


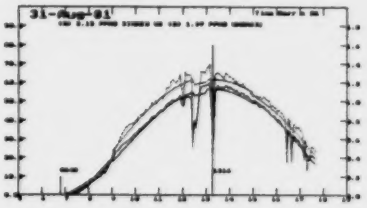


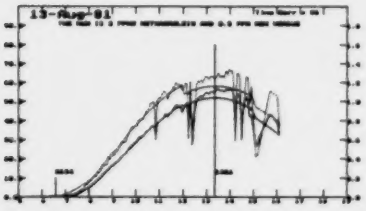
Cloudy Model

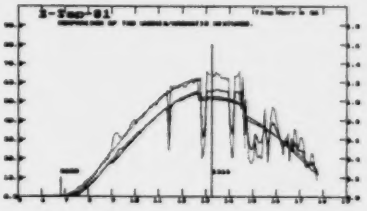


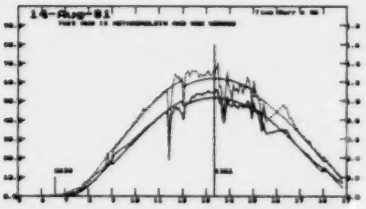


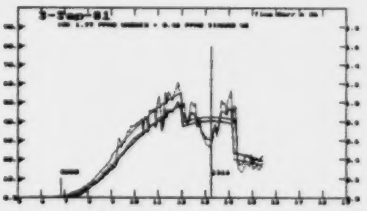


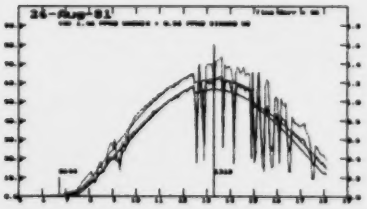


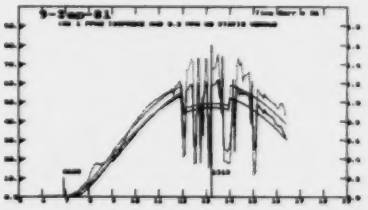


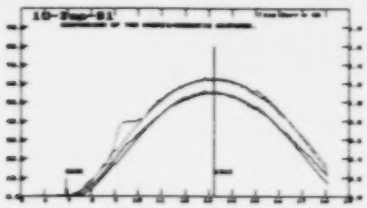


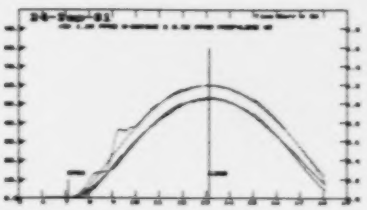


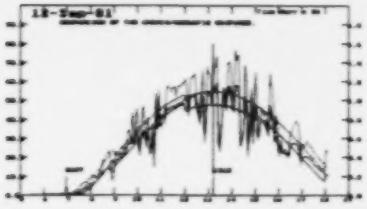


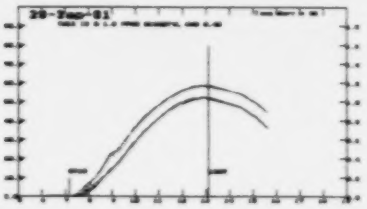


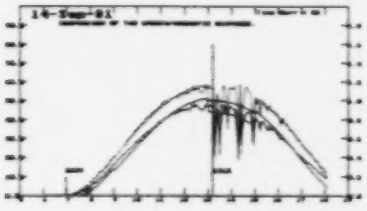


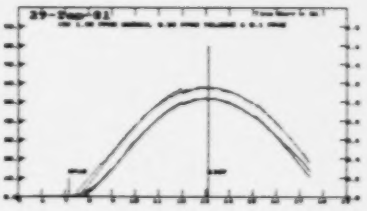


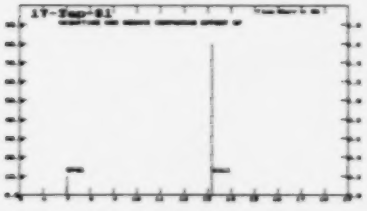


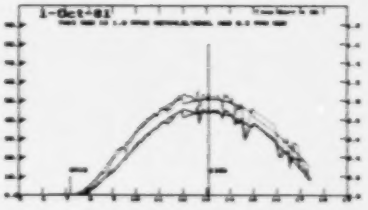


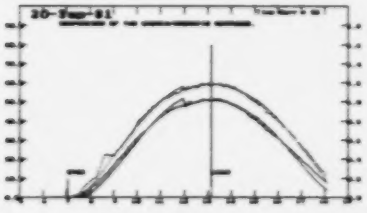


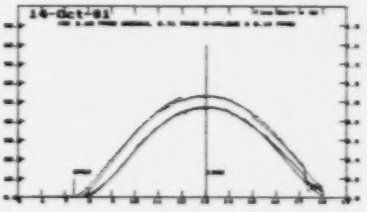




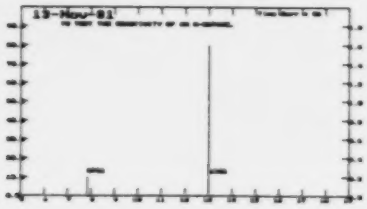


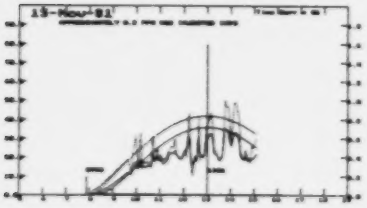


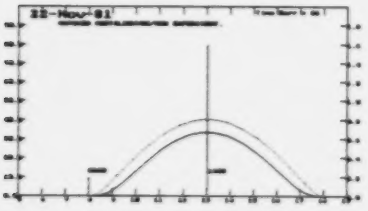


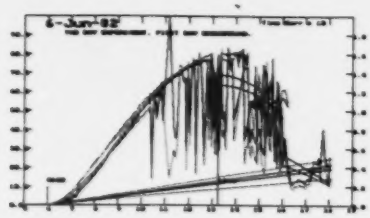


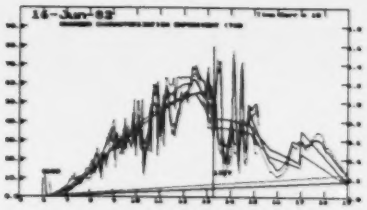
Cloudy Model

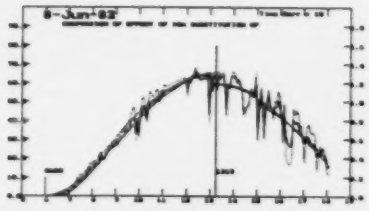


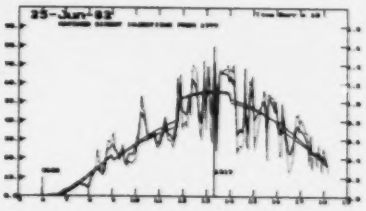


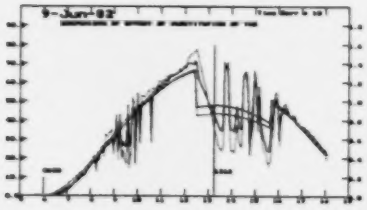


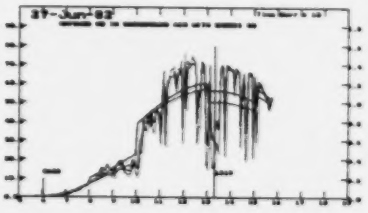


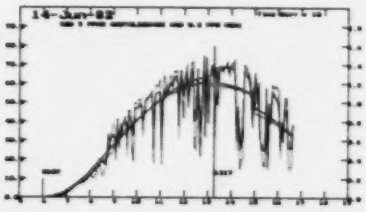


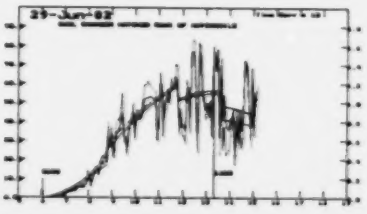


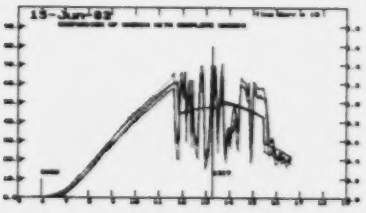


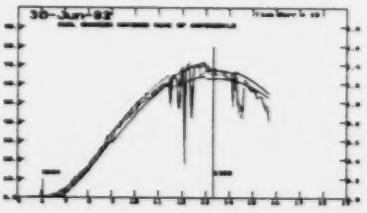




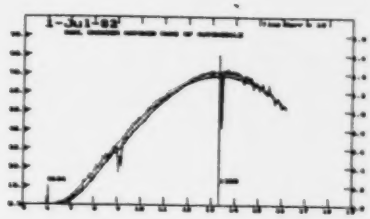


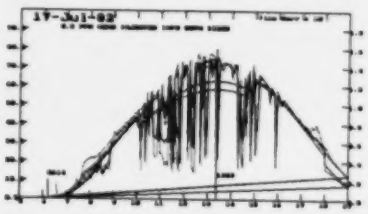


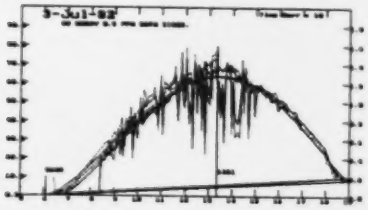


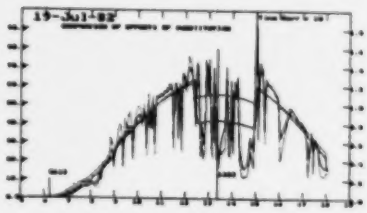


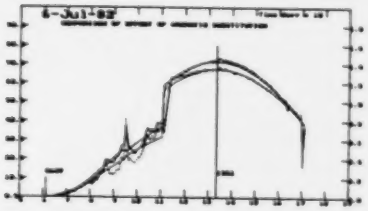
Cloudy Model

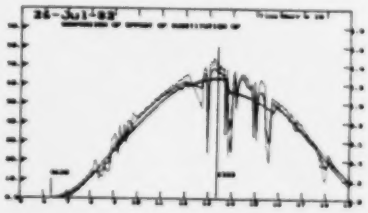


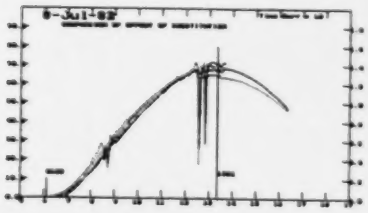


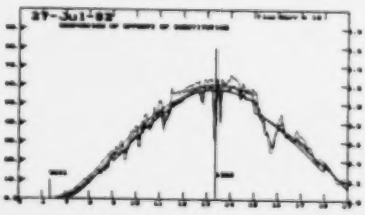


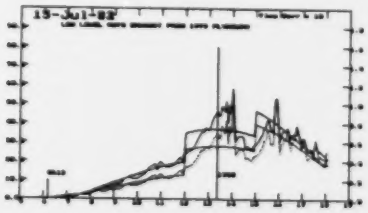


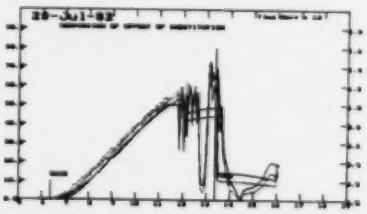


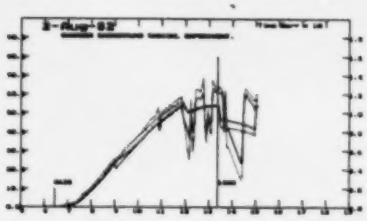
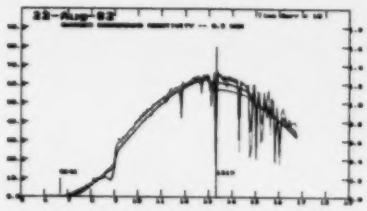
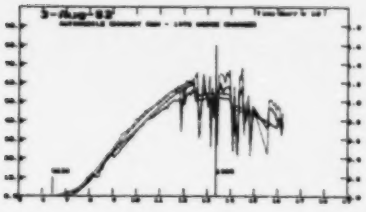
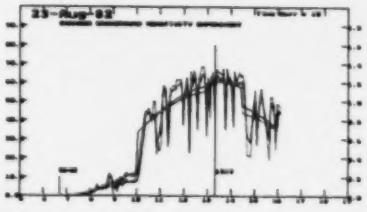
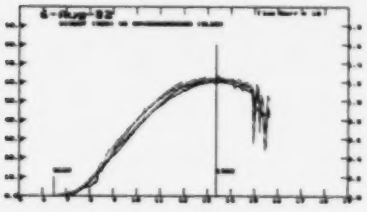
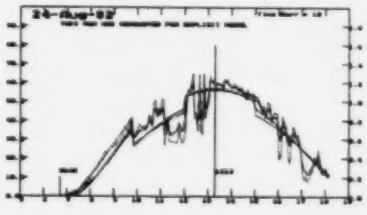
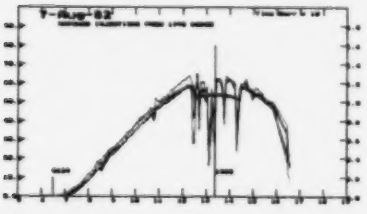
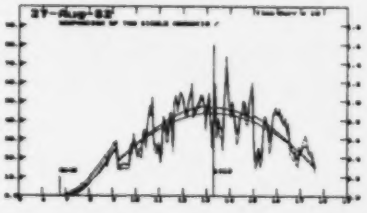
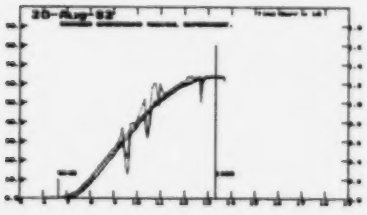
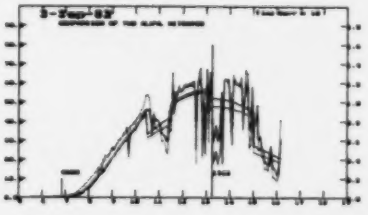


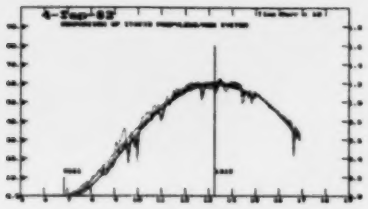


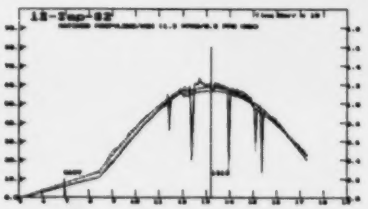


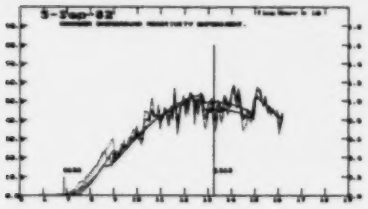


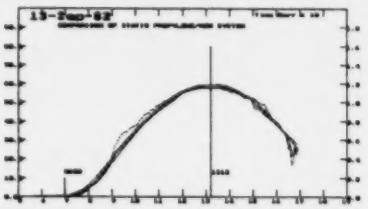


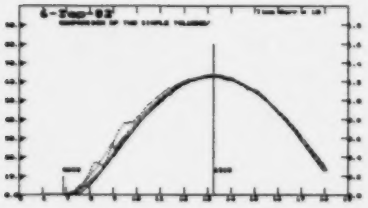











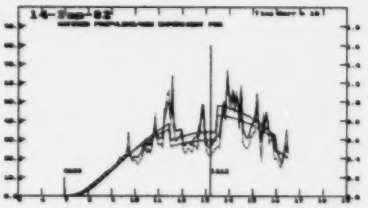


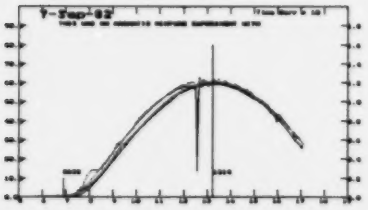


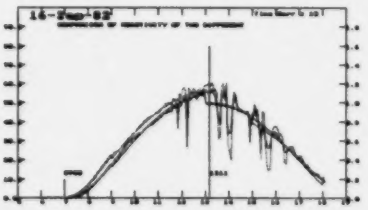


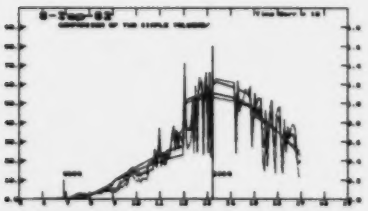


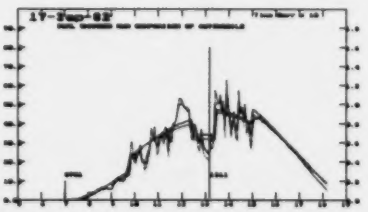




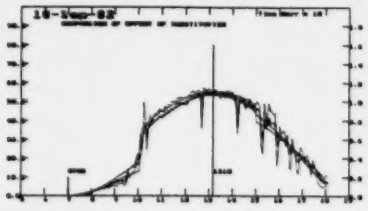


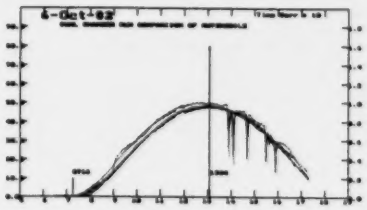


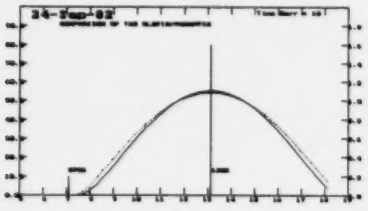


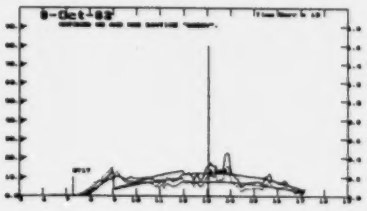


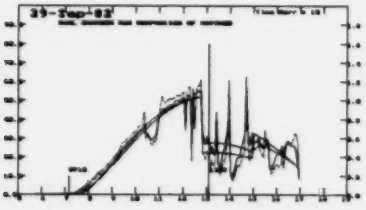
Cloudy Model

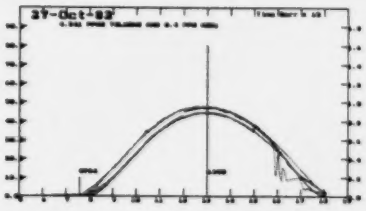


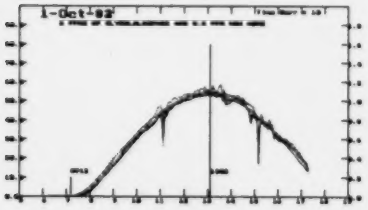


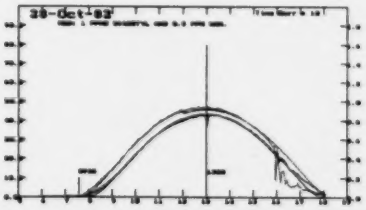


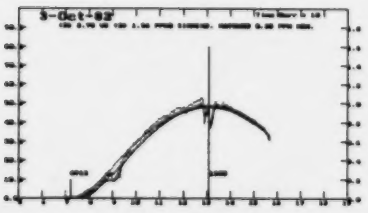


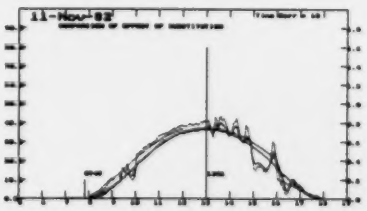




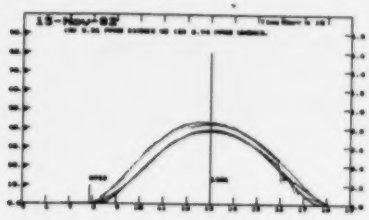


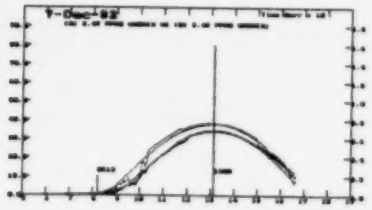




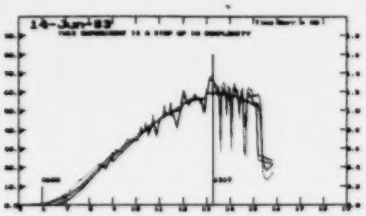


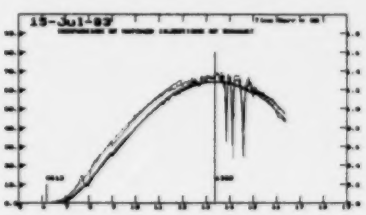
Cloudy Model

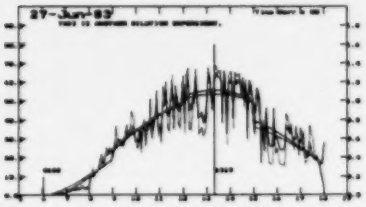


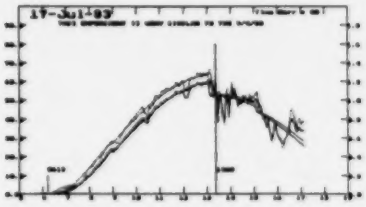


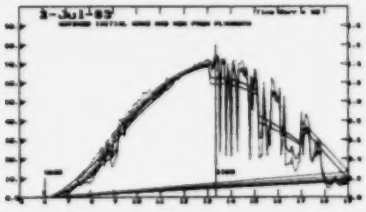
Cloudy Model

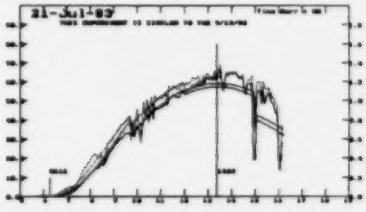


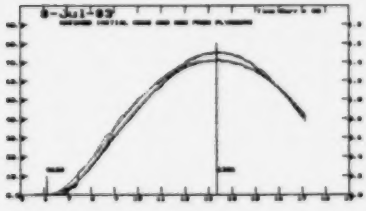


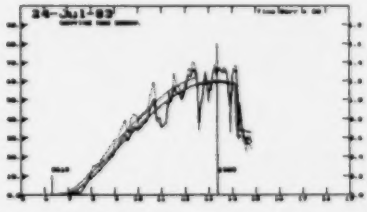


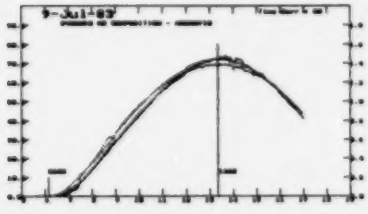


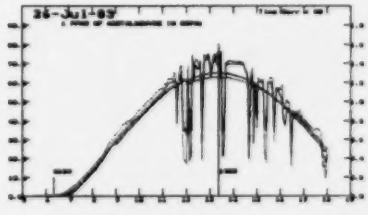




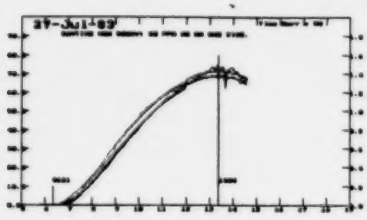


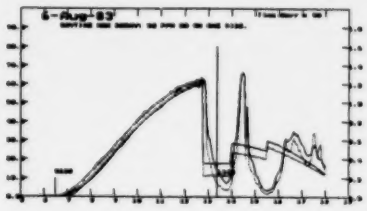


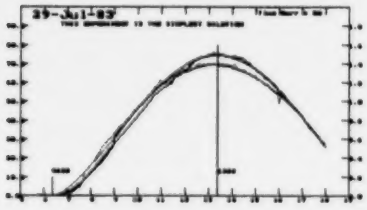


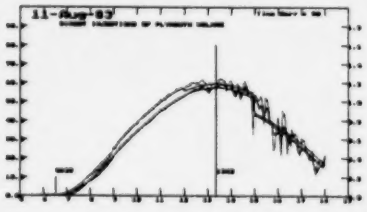


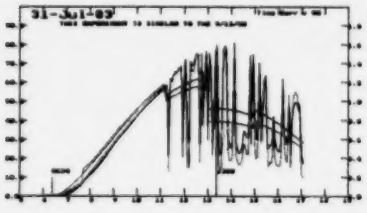
Cloudy Model

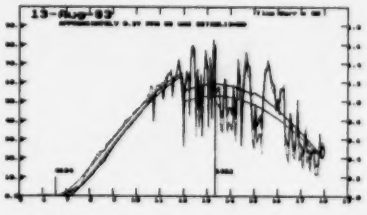


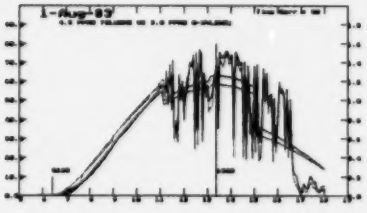


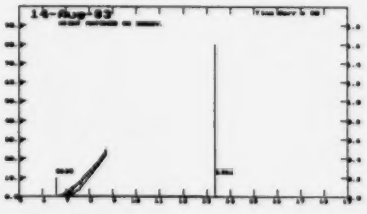


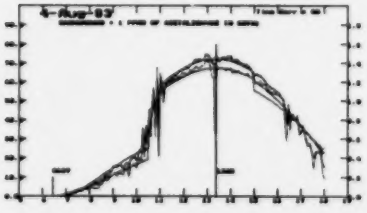


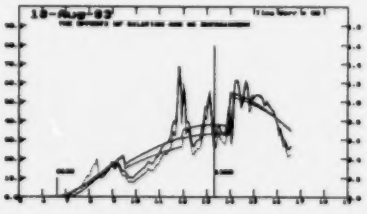


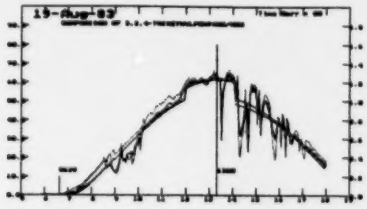


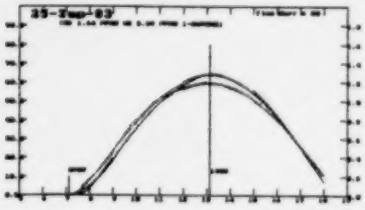


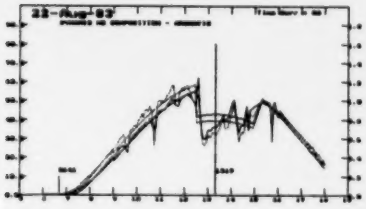


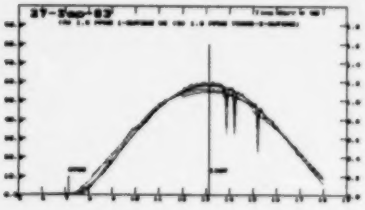


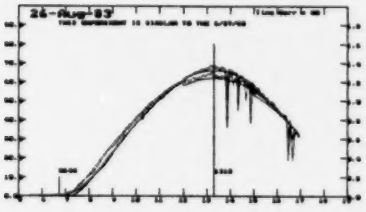


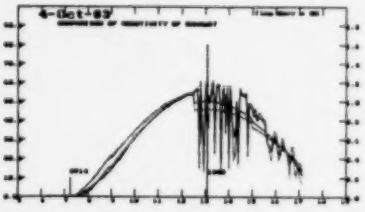


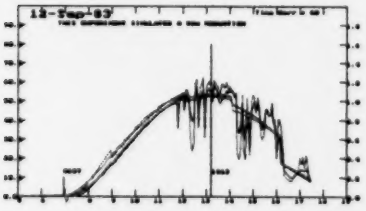


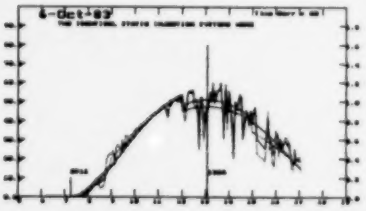


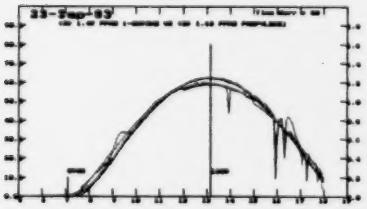


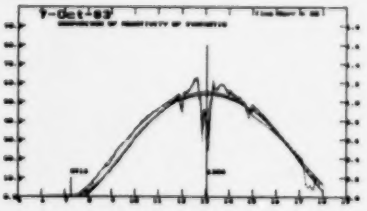




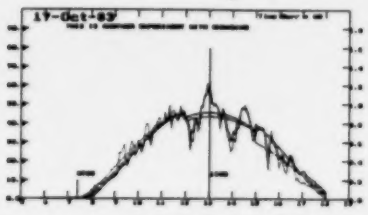




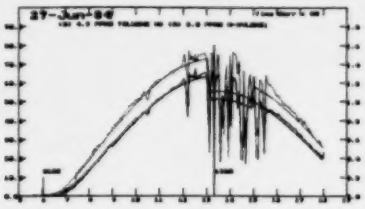


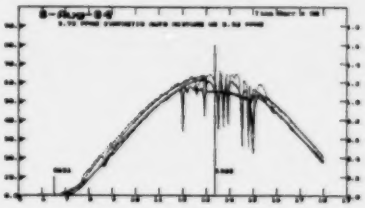


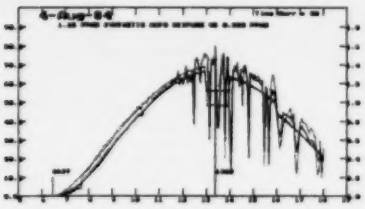
Cloudy Model

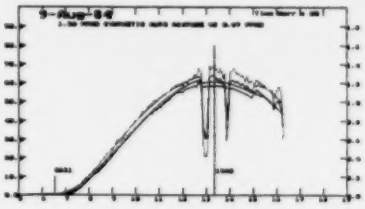


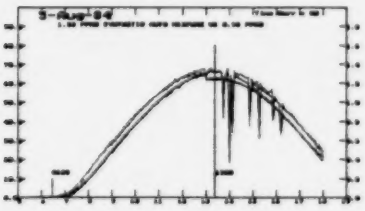
Cloudy Model

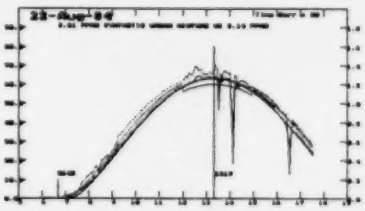


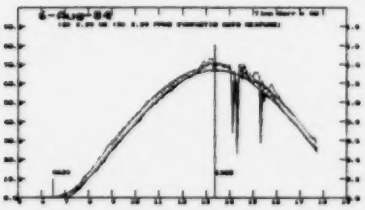


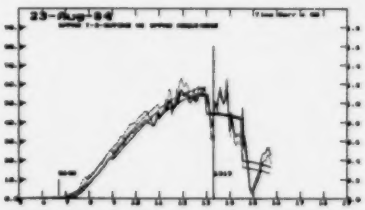


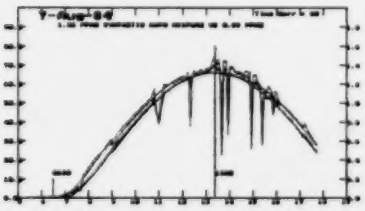


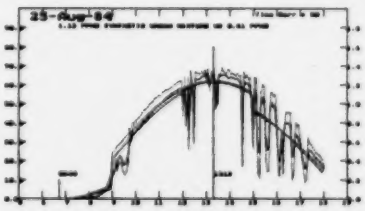




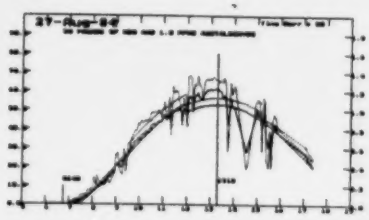


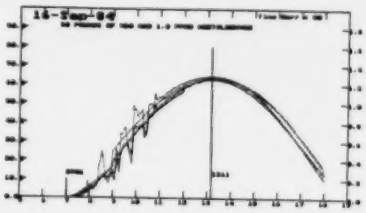


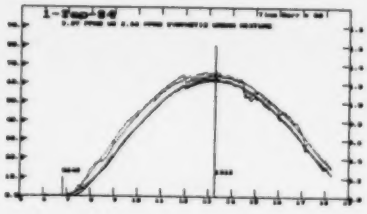


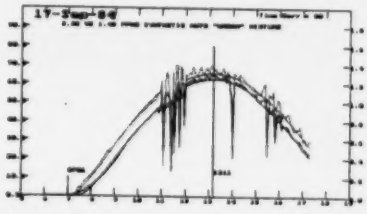


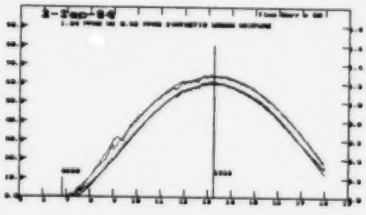
Cloudy Model

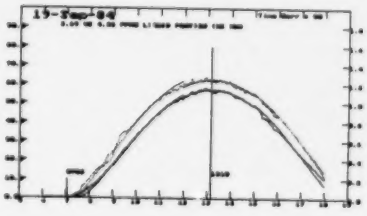


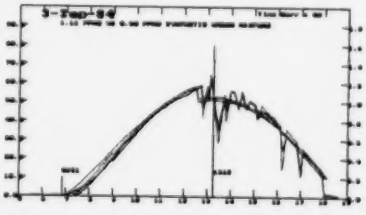


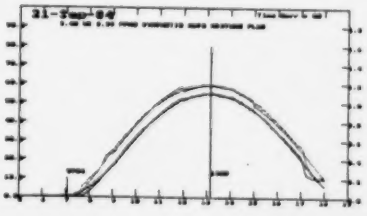


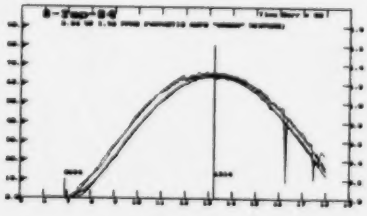


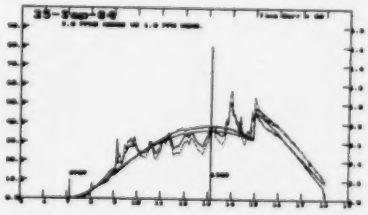




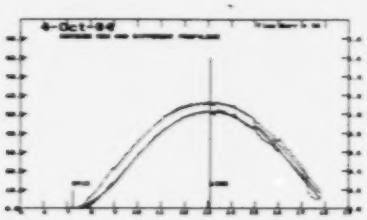


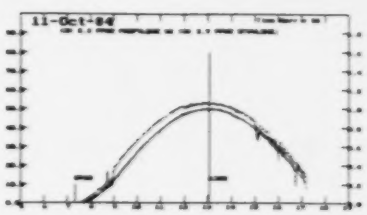


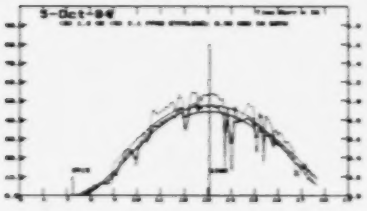


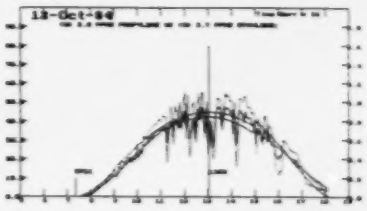


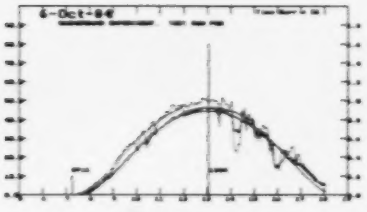
Cloudy Model

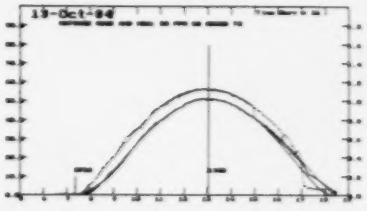


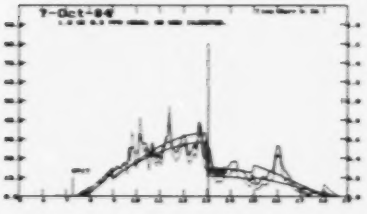


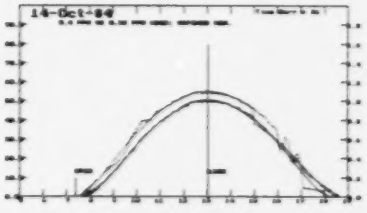


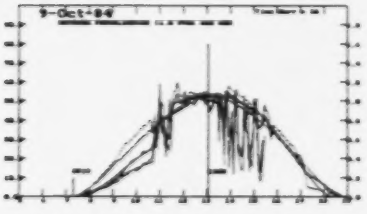


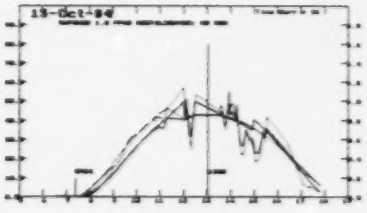




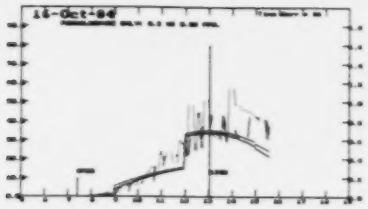


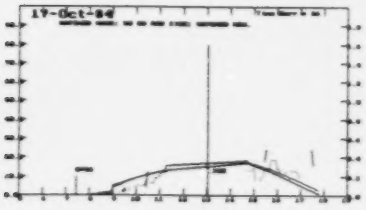


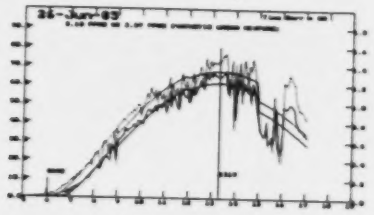


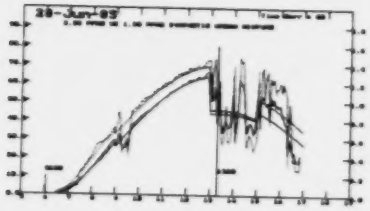


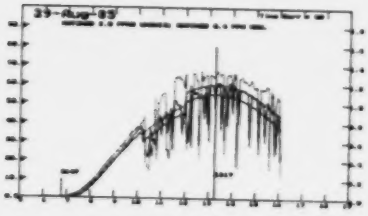
Cloudy Model

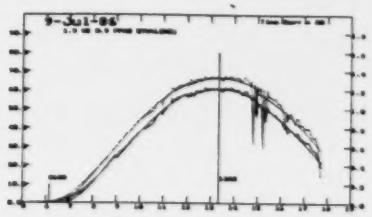


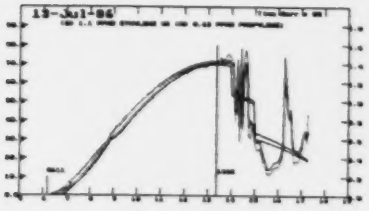




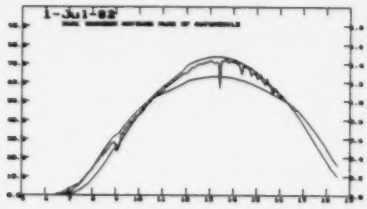


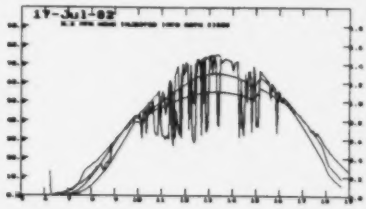


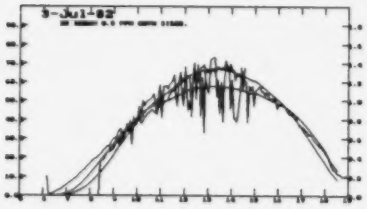


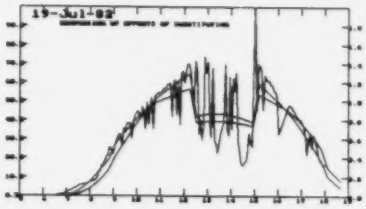


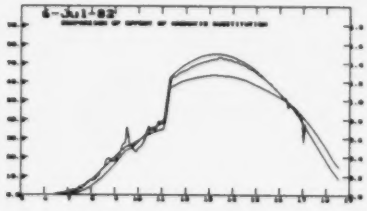
Sample In-Chamber NO_2 and HCHO Rates Plots

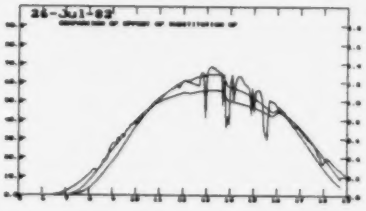


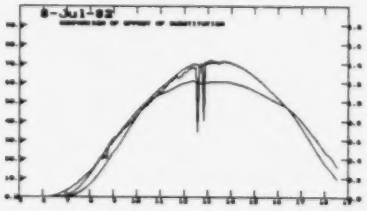


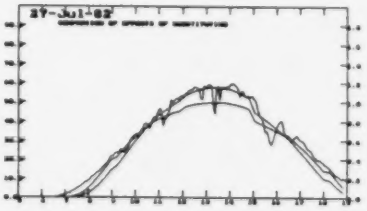


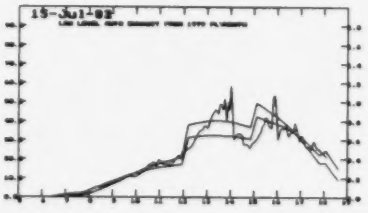


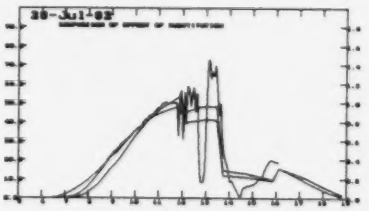


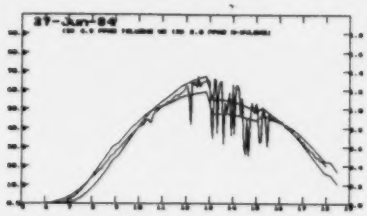


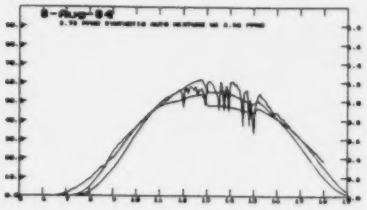


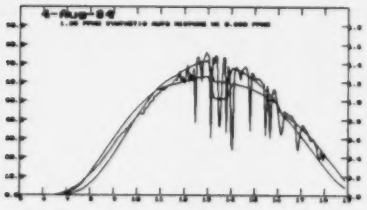


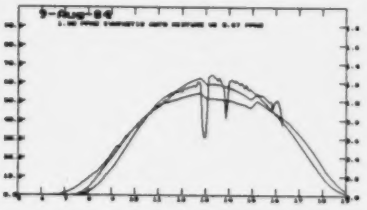


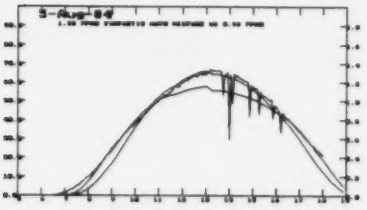


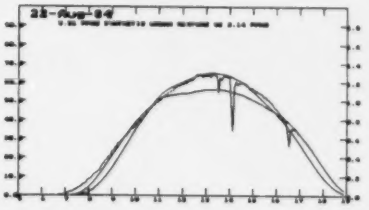


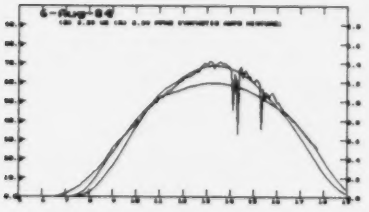


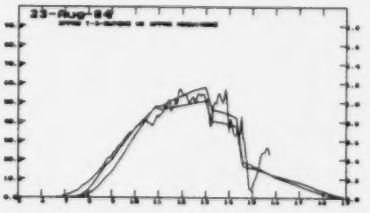


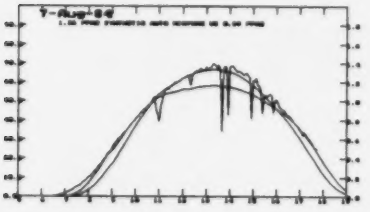


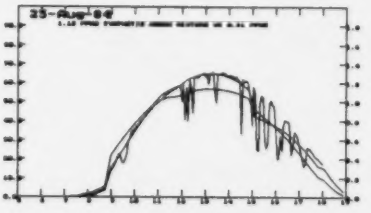


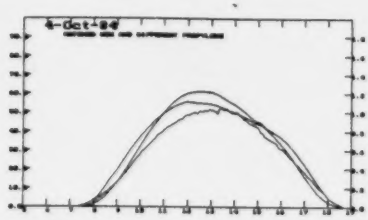


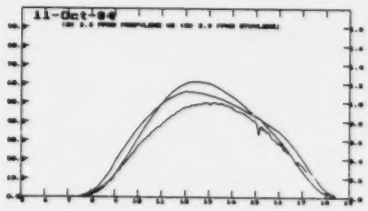


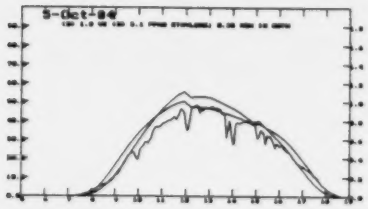


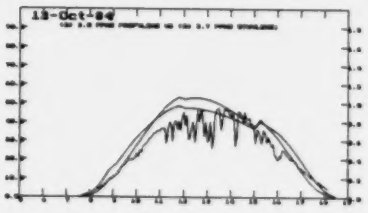


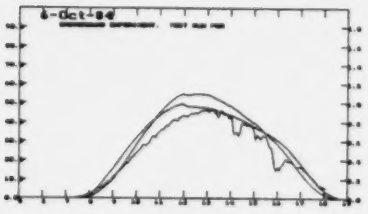


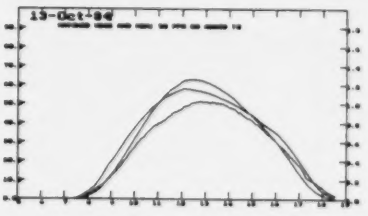


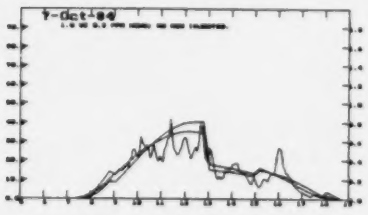


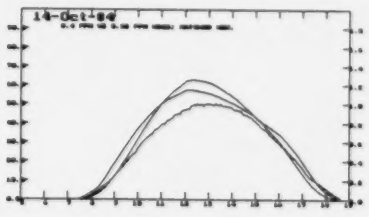


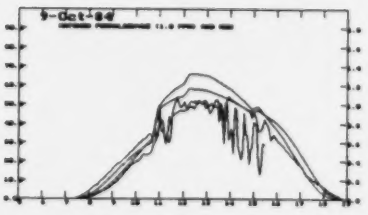


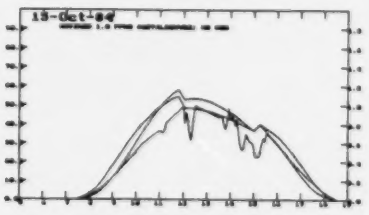












C

CsQy Files

Nitrogen Dioxide

```
NO2
! Bass,Ledford,Laufer, NBS 1976,  1/8 NM points avg to 1nm ending intervals
! Garnder, Sperry, Calvert (1987) QY
!  interpolated to be ending interval values
Ending 1nm average intervals
! Units are (cm 2/molecule)*1E-20
FAC=1E-20
298  13.0762   0.9975    309  17.0212   0.9953
299  11.8406   0.9973    310  18.5213   0.9951
300  12.3244   0.9971    311  18.4694   0.9949
301  12.0737   0.9969    312  19.5206   0.9947
302  13.0038   0.9967    313  20.6831   0.9945
303  14.6137   0.9965    314  19.6344   0.9943
304  15.4912   0.9963    315  21.1694   0.9941
305  15.3719   0.9961    316  21.7706   0.9939
306  15.4137   0.9959    317  21.7594   0.9937
307  15.4850   0.9957    318  23.6799   0.9935
308  16.7306   0.9955    319  23.4193   0.9933
                         320  24.3282   0.9931
```

Nitrogen Dioxide, cont

321	26.0206	0.9929	346	41.0781	0.9879	371	51.6187	0.9806
322	25.7632	0.9927	347	44.0630	0.9877	372	55.6838	0.9802
323	27.0232	0.9925	348	43.2169	0.9875	373	58.6765	0.9799
324	26.1693	0.9923	349	50.1038	0.9873	374	53.8809	0.9795
325	27.2432	0.9921	350	41.1461	0.9871	375	52.0286	0.9794
326	29.3356	0.9919	351	43.4794	0.9869	376	58.2527	0.9786
327	29.3450	0.9917	352	44.3613	0.9867	377	62.1152	0.9778
328	29.6781	0.9915	353	43.3586	0.9865	378	52.9526	0.9770
329	29.0938	0.9913	354	43.9583	0.9863	379	52.2119	0.9762
330	30.5056	0.9911	355	53.9105	0.9862	380	59.8696	0.9754
331	30.0505	0.9909	356	47.4056	0.9858	381	58.6157	0.9745
332	31.1738	0.9907	357	49.4844	0.9855	382	56.2312	0.9735
333	34.6706	0.9905	358	54.1505	0.9851	383	54.0598	0.9725
334	33.9006	0.9903	359	48.4862	0.9848	384	55.7454	0.9715
335	31.8175	0.9901	360	45.5769	0.9844	385	59.9087	0.9700
336	33.8964	0.9899	361	49.9193	0.9841	386	58.8442	0.9680
337	34.8694	0.9897	362	51.2462	0.9837	387	55.1511	0.9665
338	33.9031	0.9895	363	50.8627	0.9834	388	59.2727	0.9650
339	37.7600	0.9893	364	49.2869	0.9830	389	57.9805	0.9630
340	38.4726	0.9891	365	53.8564	0.9827	390	60.0955	0.9610
341	38.7887	0.9889	366	58.1191	0.9823	391	59.6780	0.9595
342	41.6575	0.9887	367	50.3062	0.9820	392	62.2339	0.9580
343	36.4994	0.9885	368	54.5300	0.9816	393	59.9294	0.9550
344	38.1173	0.9883	369	52.6089	0.9813	394	54.3645	0.9515
345	39.8107	0.9881	370	52.8513	0.9809	395	54.4453	0.9460

396	62.2927	0.9320	411	59.0000	0.1200
397	56.7043	0.8960	412	59.8200	0.1020
398	57.5889	0.8450	413	60.8800	0.0885
399	63.9028	0.7900	414	61.9400	0.0765
400	59.9580	0.7275	415	63.0000	0.0645
401	65.4565	0.6650	416	62.8600	0.0535
402	61.4414	0.5975	417	62.7200	0.0435
403	53.1899	0.5225	418	62.5800	0.0345
404	55.4019	0.4550	419	62.4400	0.0265
405	58.9150	0.3875	420	62.3000	0.0205
406	59.0605	0.3200	421	61.8400	0.0150
407	50.9722	0.2575	422	61.3800	0.0100
408	53.1895	0.2050	423	60.9200	0.0060
409	64.9121	0.1690	424	60.4600	0.0020
410	59.4600	0.1415	425	60.0000	0.0000

Nitrous Acid

```
HONO
! These data supplied by Carter on floppy 9/87
! His source was Table 5, p327 Atkinson and Lloyd which are the values
!   Stockwell and Calvert, 1978 and now recommended by NASA (1988).
Points
! Units (cm 2/molecule)
!
0.311 0.00E-20   0.331 8.70E-20   0.351 17.4E-20   0.371 9.46E-20
0.312 0.20E-20   0.332 13.8E-20   0.352 11.4E-20   0.372 8.85E-20
0.313 0.42E-20   0.333 5.91E-20   0.353 37.1E-20   0.373 7.44E-20
0.314 0.46E-20   0.334 5.91E-20   0.354 49.6E-20   0.374 4.77E-20
0.315 0.42E-20   0.335 6.45E-20   0.355 24.6E-20   0.375 2.70E-20
0.316 0.30E-20   0.336 5.91E-20   0.356 11.9E-20   0.376 1.90E-20
0.317 0.46E-20   0.337 4.58E-20   0.357 9.35E-20   0.377 1.50E-20
0.318 3.60E-20   0.338 19.1E-20   0.358 7.78E-20   0.378 1.90E-20
0.319 6.10E-20   0.339 16.3E-20   0.359 7.29E-20   0.379 5.80E-20
0.320 2.10E-20   0.340 10.5E-20   0.360 6.83E-20   0.380 7.78E-20
0.321 4.27E-20   0.341 8.70E-20   0.361 6.90E-20   0.381 11.4E-20
0.322 4.01E-20   0.342 33.5E-20   0.362 7.32E-20   0.382 14.0E-20
0.323 3.93E-20   0.343 20.1E-20   0.363 9.00E-20   0.383 17.2E-20
0.324 4.01E-20   0.344 10.2E-20   0.364 12.1E-20   0.384 19.9E-20
0.325 4.04E-20   0.345 8.54E-20   0.365 13.3E-20   0.385 19.0E-20
0.326 3.13E-20   0.346 8.32E-20   0.366 21.3E-20   0.386 11.9E-20
0.327 4.12E-20   0.347 8.20E-20   0.367 35.2E-20   0.387 5.65E-20
0.328 7.55E-20   0.348 7.49E-20   0.368 45.0E-20   0.388 3.20E-20
0.329 6.64E-20   0.349 7.13E-20   0.369 29.3E-20   0.389 1.90E-20
0.330 7.29E-20   0.350 6.83E-20   0.370 11.9E-20   0.390 1.20E-20
                                                  0.391 0.50E-20
                                                  0.392 0.00E-20
```

Nitrate Radical

```
NO3=NO
! These data supplied by Carter on floppy 9/87 and in June 1988
! The NO3 absorption coefficients and quantum yields are based on
! the evalution of Carter et al (1986).
! [NASA(1988) recommendation could not be used because it gives no
!  absorption coefficients below 600 nm, and and makes no
!  recommendation for the quantum yields.]
! Absorption cross sections discussed by Atkinson and Lloyd (1984).
! Quantum yields of Magnotta and Johnson (1980), divided by a factor
!  of 1.5 to yield unit quantum yields at 580-590 nm, used.
Centered
!      Based upon inspection of Table 4 Atk&Ll and the values below.
! Units (cm 2/molecule)
 0.580  0.0       0.00
 0.585  2.770E-18 0.00
 0.590  5.140E-18 0.25
 0.595  4.080E-18 0.40
 0.600  2.830E-18 0.25
 0.605  3.450E-18 0.20
 0.610  1.480E-18 0.20
 0.615  1.960E-18 0.10
 0.620  3.580E-18 0.10
 0.625  9.250E-18 0.05
 0.630  5.660E-18 0.05
 0.635  1.450E-18 0.03
 0.640  1.110E-18 0.00
```

Nitrate Radical to NO_2

```
NO3=NO2
! These data supplied by Carter on floppy 9/87 and June 1988
! Based on the evaluation of Carter et al (1986).
! See Comments in NO3NO file.
Centered
!       based upon inspection of Table 4 Atkinson and Lloyd
! Units (cm 2/molecule)
0.400  0.000E-19 1.000     0.520  1.450E-18 1.000
0.405  0.300E-19 1.000     0.525  1.480E-18 1.000
0.410  0.400E-19 1.000     0.530  1.940E-18 1.000
0.415  0.500E-19 1.000     0.535  2.040E-18 1.000
0.420  0.800E-19 1.000     0.540  1.810E-18 1.000
0.425  1.000E-19 1.000     0.545  1.810E-18 1.000
0.430  1.300E-19 1.000     0.550  2.360E-18 1.000
0.435  1.800E-19 1.000     0.555  2.680E-18 1.000
0.440  1.900E-19 1.000     0.560  3.070E-18 1.000
0.445  2.200E-19 1.000     0.565  2.530E-18 1.000
0.450  2.800E-19 1.000     0.570  2.540E-18 1.000
0.455  3.300E-19 1.000     0.575  2.740E-18 1.000
0.460  3.700E-19 1.000     0.580  3.050E-18 1.000
0.465  4.300E-19 1.000     0.585  2.770E-18 1.000
0.470  5.100E-19 1.000     0.590  5.140E-18 0.750
0.475  6.000E-19 1.000     0.595  4.080E-18 0.600
0.480  6.400E-19 1.000     0.600  2.830E-18 0.550
0.485  6.900E-19 1.000     0.605  3.450E-18 0.400
0.490  8.800E-19 1.000     0.610  1.450E-18 0.300
0.495  9.500E-19 1.000     0.615  1.960E-18 0.250
0.500  1.010E-18 1.000     0.620  3.580E-18 0.200
0.505  1.100E-18 1.000     0.625  9.250E-18 0.150
0.510  1.320E-18 1.000     0.630  5.660E-18 0.050
0.515  1.400E-18 1.000     0.635  1.450E-18 0.000
```

Ozone, UV

```
O3toO1D
! Bass and Paur (1984) 1nm averages from high resolution data
! QY are from formula on p322 Atkinson and Lloyd at a temp of 300K.
Centered
! With FAC, units are (cm 2/molecule) x 1000
FAC=1.0E-17
280 397.0   0.900    298 50.654  0.900    316   4.610   0.081
281 360.0   0.900    299 45.187  0.900    317   4.170   0.048
282 324.0   0.900    300 39.245  0.900    318   3.720   0.022
283 301.0   0.900    301 34.245  0.900    319   2.690   0.002
284 273.0   0.900    302 30.547  0.900    320   3.232   0.000
285 244.0   0.900    303 26.011  0.900
286 221.0   0.900    304 23.741  0.900
287 201.0   0.900    305 20.112  0.884
288 176.0   0.900    306 17.870  0.848
289 158.0   0.900    307 15.600  0.800
290 141.0   0.900    308 13.760  0.738
291 126.0   0.900    309 12.520  0.657
292 110.0   0.900    310 10.200  0.558
293 98.90   0.900    311  9.170  0.448
294 85.942  0.900    312  7.880  0.344
295 76.968  0.900    313  6.770  0.250
296 66.677  0.900    314  6.350  0.178
297 58.414  0.900    315  5.099  0.123
```

Ozone, visible

```
O3=O(3P)
! These data supplied by Carter June 1988.
!       Absorption coefficients:  280-320 from Bass and Paur (1984).
!       Data for wl > 320 that used by Carter et al (1986) EPA report.
!       Quantum yields calculated from those for O1D path, assuming
!       that the total quantum yield is 1.
End Intervals
!     based upon O1D averaging method
! Units (cm 2/molecule) then multiplied by 1000
FAC=1000.0
0.280 3.970E-18  0.100   0.301 3.424E-19  0.100   0.340 1.700E-21  1.000
0.281 3.600E-18  0.100   0.302 3.055E-19  0.100   0.350 4.000E-22  1.000
0.282 3.240E-18  0.100   0.303 2.601E-19  0.100   0.355 0.000E-01  1.000
0.283 3.010E-18  0.100   0.304 2.374E-19  0.100   0.400 0.000E-01  1.000
0.284 2.730E-18  0.100   0.305 2.011E-19  0.112   0.450 1.600E-22  1.000
0.285 2.440E-18  0.100   0.306 1.787E-19  0.149   0.500 1.340E-21  1.000
0.286 2.210E-18  0.100   0.307 1.560E-19  0.197   0.550 3.320E-21  1.000
0.287 2.010E-18  0.100   0.308 1.376E-19  0.259   0.600 5.060E-21  1.000
0.288 1.760E-18  0.100   0.309 1.252E-19  0.339   0.650 2.450E-21  1.000
0.289 1.580E-18  0.100   0.310 1.020E-19  0.437   0.700 8.700E-22  1.000
0.290 1.410E-18  0.100   0.311 9.170E-20  0.546   0.750 3.200E-22  1.000
0.291 1.260E-18  0.100   0.312 7.880E-20  0.652   0.800 1.600E-22  1.000
0.292 1.100E-18  0.100   0.313 6.770E-20  0.743   0.900 0.000E-00  1.000
0.293 9.890E-19  0.100   0.314 6.350E-20  0.816
0.294 8.594E-19  0.100   0.315 5.099E-20  0.872
0.295 7.697E-19  0.100   0.316 4.610E-20  0.916
0.296 6.668E-19  0.100   0.317 4.170E-20  0.949
0.297 5.841E-19  0.100   0.318 3.720E-20  0.976
0.298 5.065E-19  0.100   0.319 2.690E-20  0.997
0.299 4.519E-19  0.100   0.320 3.232E-20  1.000
0.300 3.924E-19  0.100   0.330 6.700E-21  1.000
```

Hydrogen Peroxide

```
H2O2
! These data supplied by Carter on floppy 9/87 and June 1988
! His source was Table 7 p334, Atkinson and Lloyd which were based on
!  mean of Molina and Molina, 1981  and Lin et al, 1978.
! These are also the NASA(1988) and NASA(1983) recommendations.
! QY is assumed to be 1.0.
Points
!    assumed from Carter's treatment, not sure
! Units are (cm 2/molecule) multiplied by 1000 for display
FAC=1000.0
0.250  8.30E-20    0.305  0.49E-20
0.255  6.70E-20    0.310  0.37E-20
0.260  5.20E-20    0.315  0.28E-20
0.265  4.20E-20    0.320  0.20E-20
0.270  3.20E-20    0.325  0.15E-20
0.275  2.50E-20    0.330  0.12E-20
0.280  2.00E-20    0.335  0.09E-20
0.285  1.50E-20    0.340  0.07E-20
0.290  1.13E-20    0.345  0.05E-20
0.295  0.87E-20    0.350  0.03E-20
0.300  0.66E-20    0.355  0.00E-20
```

Hydro-Methyl Peroxide

```
CO2H
!Hydro-Methyl peroxide
! These data were supplied by Carter on floppy 9/87.
!  His source was MOLINA & ARGUELLO, GEOPHYS.RES.LETT 6,953 (1979)
Points
!    ^ assumed from way Carter's program treated these data.
! Units (cm^2/molecule) multiplied by 1000 for display
FAC=1000.0
 0.210  3.750E-19 1.000
 0.220  2.200E-19 1.000
 0.230  1.380E-19 1.000
 0.240  8.800E-20 1.000
 0.250  5.800E-20 1.000
 0.260  3.800E-20 1.000
 0.270  2.500E-20 1.000
 0.280  1.500E-20 1.000
 0.290  9.000E-21 1.000
 0.300  5.800E-21 1.000
 0.310  3.400E-21 1.000
 0.320  1.900E-21 1.000
 0.330  1.100E-21 1.000
 0.340  6.000E-22 1.000
 0.350  4.000E-22 1.000
 0.360  0.000E-01 1.000
```

Formaldehyde to Radicals

```
HCHOr
! These values from Mike Gery, Table 2-4, EPA Internal Report
!    "An Assessment of Existing UV Radiation Data Used in AQ Modeling"
! They are Bass 0.05 nm cs averged to 1nm intervals.
Centered
! ^assumed
! Units are (cm^2/molecule)*1E-20 further multiplied by 1000
FAC=1E-17
```

279	0.0	0.0	311	0.81	0.75
280	2.34	0.56	312	1.49	0.74
281	1.65	0.58	313	1.55	0.73
282	0.76	0.6	314	3.99	0.72
283	0.46	0.62	315	2.88	0.7
284	3.93	0.63	316	2.79	0.69
285	3.46	0.65	317	3.59	0.67
286	2.32	0.67	318	1.65	0.65
287	0.95	0.68	319	0.73	0.63
288	2.32	0.7	320	1.71	0.61
289	2.5	0.71	321	1.32	0.59
290	1.43	0.72	322	0.43	0.57
291	1.32	0.73	323	0.6	0.54
292	0.66	0.75	324	0.75	0.51
293	5.22	0.76	325	2.19	0.49
294	4.3	0.76	326	3.44	0.46
295	3.21	0.77	327	1.75	0.43
296	1.59	0.78	328	1.01	0.39
297	1.96	0.79	329	3.03	0.36
298	3.66	0.79	330	1.96	0.33
299	1.55	0.79	331	0.79	0.29
300	0.72	0.8	332	0.32	0.25
301	1.51	0.8	333	0.15	0.21
302	0.74	0.8	334	0.17	0.17
303	4.35	0.8	335	0.02	0.13
304	4.79	0.8	336	0.17	0.083
305	4.94	0.79	337	0.32	0.038
306	3.02	0.79	338	1.93	0.0
307	1.16	0.79	339	2.15	0.0
308	2.18	0.78	340	1.07	0.0
309	2.25	0.77			
310	1.03	0.76			

Formaldehyde to Stable Products

```
HCHOs
! These values from Mike Gery, Table 2-4, EPA Internal Report
!    "An Assessment of Existing UV Radiation Data Used in AQ Modeling"
! They are Bass 0.05 nm cs averged to 1nm intervals.
Centered
! ^assumed
! Units are (cm^2/molecule)*1E-20 further multiplied by 1000
FAC=1E-17
```

279	0.0	0.0	320	1.71	0.39
280	2.34	0.44	321	1.32	0.41
281	1.65	0.42	322	0.43	0.43
282	0.76	0.4	323	0.6	0.46
283	0.46	0.38	324	0.75	0.49
284	3.93	0.37	325	2.19	0.51
285	3.46	0.35	326	3.44	0.54
286	2.32	0.33	327	1.75	0.55
287	0.95	0.32	328	1.01	0.57
288	2.32	0.3	329	3.03	0.58
289	2.5	0.29	330	1.96	0.59
290	1.43	0.28	331	0.79	0.6
291	1.32	0.27	332	0.32	0.61
292	0.66	0.25	333	0.15	0.62
293	5.22	0.24	334	0.17	0.62
294	4.3	0.24	335	0.02	0.62
295	3.21	0.23	336	0.17	0.62
296	1.59	0.22	337	0.32	0.62
297	1.96	0.21	338	1.93	0.61
298	3.66	0.21	339	2.15	0.61
299	1.55	0.21	340	1.07	0.6
300	0.72	0.2	341	0.31	0.59
301	1.51	0.2	342	0.94	0.57
302	0.74	0.2	343	1.37	0.56
303	4.35	0.2	344	0.57	0.54
304	4.79	0.2	345	0.12	0.52
305	4.94	0.21	346	0.04	0.5
306	3.02	0.21	347	0.04	0.47
307	1.16	0.21	348	0.07	0.45
308	2.18	0.22	349	0.03	0.42
309	2.25	0.23	350	0.03	0.39
310	1.03	0.24	351	0.09	0.36
311	0.81	0.25	352	0.9	0.33
312	1.49	0.26	353	1.17	0.3
313	1.55	0.27	354	0.72	0.26
314	3.99	0.28	355	0.26	0.23
315	2.88	0.3	356	0.05	0.2
316	2.79	0.31	357	0.03	0.16
317	3.59	0.33	358	0.04	0.13
318	1.65	0.35	359	0.03	0.1
319	0.73	0.37			

Acetaldehyde to Radicals

```
CCHO
! This file came from Carter by floppy on 9/87 as part of his
!  photolysis program.
! Values for QY are interpolated from Table 10 p340 Atkinson and Lloyd
!   and are based upon Horowitz and Calvert, 1982 and
!   Meyrahn and Moortgat, 1982.
Points
! ^ based on footnote on Table 10.
! Units (cm^2/molecule) then multiplied by 1000 for display
FAC=1000.0
 0.260  2.160E-20 0.310
 0.270  3.460E-20 0.380
 0.280  4.320E-20 0.590
 0.290  4.760E-20 0.550
 0.295  4.450E-20 0.490
 0.300  4.030E-20 0.415
 0.305  3.420E-20 0.325
 0.310  2.720E-20 0.235
 0.315  2.090E-20 0.150
 0.320  1.600E-20 0.075
 0.325  1.130E-20 0.030
 0.330  6.800E-21 0.000
 0.335  0.0       0.0
```

Propyaldehyde to Radicals

```
C2CHO
! For the CAL reaction:
!   RCHO + HV = C2-O2. + RO2. + CO + HO2.
Centered
!units are (cm^2/molecule) further multiplied by 1000
!
FAC=1000.0
0.270  0.00E+00  0.0
0.280  5.26E-20  0.96
0.290  5.77E-20  0.91
0.300  5.05E-20  0.86
0.310  3.68E-20  0.60
0.320  1.66E-20  0.36
0.330  6.49E-21  0.20
0.340  1.44E-21  0.08
0.345  0.0       0.02
```

Benzaldehyde

```
BZCHO
!Benzaldehyde in N-hexane
! These data were also supplied by Carter on Floppy, 9/87.
! Carter supplied the QY
! Majer, Naman, and Robb,  Trans. Faraday Soc, 1969 p. 1846
! Carter's quote: "Crude.  Taken from sloppy-looking plot."
! For CAL reaction
!   BZ-CHO + HV = #7 -C
Points
!units are (cm^2/molecule) multiplied by 1000 for display
FAC=1000.0
0.299       1.776E-19 0.05
0.304       7.40E-20  0.05
0.306       6.91E-20  0.05
0.309       6.41E-20  0.05
0.313       6.91E-20  0.05
0.314       6.91E-20  0.05
0.318       6.41E-20  0.05
0.325       8.39E-20  0.05
0.332       7.65E-20  0.05
0.338       8.88E-20  0.05
0.342       8.88E-20  0.05
0.346       7.89E-20  0.05
0.349       7.89E-20  0.05
0.354       9.13E-20  0.05
0.355       8.14E-20  0.05
0.364       5.67E-20  0.05
0.368       6.66E-20  0.05
0.369       8.39E-20  0.05
0.370       8.39E-20  0.05
0.372       3.45E-20  0.05
0.374       3.21E-20  0.05
0.376       2.47E-20  0.05
0.377       2.47E-20  0.05
0.380       3.58E-20  0.05
0.382       0.99E-20  0.05
```

Acetone

```
ACETONE
! These data supplied by Carter June 1988
! Abs. coeffs from RADM model.  QY's of Meyrhan et al (1986).
! rxn ACETONE + HV = CH3CO + CH3.
Centered
!units are (cm^2/molecule) further multiplied by 1000
FAC=1000.0
0.2798  5.300E-20  0.56    0.3080  1.800E-20  0.07
0.2837  5.300E-20  0.46    0.3090  1.700E-20  0.06
0.2878  5.100E-20  0.36    0.3100  1.500E-20  0.05
0.2920  4.400E-20  0.25    0.3110  1.400E-20  0.048
0.2963  3.500E-20  0.21    0.3120  1.300E-20  0.046
0.3005  3.000E-20  0.15    0.3130  1.200E-20  0.043
0.3030  2.800E-20  0.12    0.3140  1.100E-20  0.041
0.3040  2.500E-20  0.11    0.3160  9.200E-21  0.037
0.3050  2.300E-20  0.10    0.3200  5.300E-21  0.028
0.3060  2.100E-20  0.09    0.3250  2.800E-21  0.031
0.3070  2.000E-20  0.08    0.3300  1.900E-21  0.033
                           0.3350  0.000E-01  0.036
```

Methyl-Ethyl-Ketone

```
MEK
!MEK ABSORPTION COEF'S FROM CALVERT AND PITTS (1966) - QY FROM CARTER
! For CAL reaction
!   MEK + HV = CCO-O2. + C2-O2. + RCO3. + RO2.
Centered
!Units are (cm^2/molecule) times 1000
!
FAC=1000
 0.210  1.100E-21 0.100
 0.220  1.200E-21 0.100
 0.230  4.600E-21 0.100
 0.240  1.300E-20 0.100
 0.250  2.680E-20 0.100
 0.260  4.210E-20 0.100
 0.270  5.540E-20 0.100
 0.280  5.920E-20 0.100
 0.290  5.160E-20 0.100
 0.300  3.440E-20 0.100
 0.310  1.530E-20 0.100
 0.320  4.600E-21 0.100
 0.330  1.100E-21 0.100
 0.340  0.000E-01 0.100
```

Glyoxal, below 350 NM

```
Gly1 <350nm
! GLYOXAL ABS COEFS(WL < 340)  PLUM ET AL,(1983)
!  New data below 340 shows different yield per Carter June 1988
!  ie, HCOCHO + HV + #1.00 = #0.8 HO2 + #0.45 HCHO + #10.55 CO
!  and he assumes QY=1.0 based upon Langford and Moore (1984)
!
Points  picked from spectal plot
!units are (cm^2/molecule) further multiplied by 1000
FAC=1000.0
  0.230  0.287E-20
  0.235  0.287E-20
  0.240  0.430E-20
  0.245  0.573E-20
  0.250  0.860E-20
  0.255  0.115E-19
  0.260  0.143E-19
  0.265  0.186E-19
  0.270  0.229E-19
  0.275  0.258E-19
  0.280  0.287E-19
  0.285  0.330E-19
  0.290  0.315E-19
  0.295  0.330E-19
  0.300  0.358E-19
  0.305  0.272E-19
  0.310  0.272E-19
  0.312  0.287E-19
  0.315  0.229E-19
  0.320  0.143E-19
  0.325  0.115E-19
  0.327  0.143E-19
  0.330  0.115E-19
  0.335  0.287E-20
  0.340  0.000E+00
```

Glyoxal, above 350 NM

```
Gly2,>350nm
! GLYOXAL ABS COEFS, WL > 340 PLUM  10/82
! For CAL reaction
!  HCOCHO + HV + #QYO.GLY = #0.13 HCHO + #0.87 H2 + #1.87 CO
Points  picked from spectral plot
!
!units are (cm^2/molecule) further multiplied by 1000
FAC=1000.0
0.355    0.000E+00  0.029      0.4225   0.105E-18  0.029
0.360    0.229E-20  0.029      0.423    0.851E-19  0.029
0.365    0.287E-20  0.029      0.424    0.608E-19  0.029
0.370    0.803E-20  0.029      0.425    0.729E-19  0.029
0.375    0.100E-19  0.029      0.426    0.118E-18  0.029
0.380    0.172E-19  0.029      0.4265   0.130E-18  0.029
0.382    0.158E-19  0.029      0.427    0.107E-18  0.029
0.384    0.149E-19  0.029      0.428    0.166E-18  0.029
0.386    0.149E-19  0.029      0.429    0.405E-19  0.029
0.388    0.287E-19  0.029      0.430    0.507E-19  0.029
0.390    0.315E-19  0.029      0.431    0.486E-19  0.029
0.391    0.324E-19  0.029      0.432    0.405E-19  0.029
0.392    0.304E-19  0.029      0.433    0.365E-19  0.029
0.393    0.223E-19  0.029      0.434    0.405E-19  0.029
0.394    0.263E-19  0.029      0.4345   0.608E-19  0.029
0.395    0.304E-19  0.029      0.435    0.507E-19  0.029
0.396    0.263E-19  0.029      0.436    0.811E-19  0.029
0.397    0.243E-19  0.029      0.4365   0.113E-18  0.029
0.398    0.324E-19  0.029      0.437    0.527E-19  0.029
0.399    0.304E-19  0.029      0.438    0.101E-18  0.029
0.400    0.284E-19  0.029      0.4385   0.138E-18  0.029
0.401    0.324E-19  0.029      0.439    0.770E-19  0.029
0.402    0.446E-19  0.029      0.440    0.247E-18  0.029
0.403    0.527E-19  0.029      0.441    0.811E-19  0.029
0.404    0.426E-19  0.029      0.442    0.608E-19  0.029
0.405    0.304E-19  0.029      0.443    0.750E-19  0.029
0.406    0.304E-19  0.029      0.444    0.932E-19  0.029
0.407    0.284E-19  0.029      0.445    0.113E-18  0.029
0.408    0.243E-19  0.029      0.446    0.527E-19  0.029
0.409    0.284E-19  0.029      0.447    0.243E-19  0.029
0.410    0.608E-19  0.029      0.448    0.284E-19  0.029
0.411    0.507E-19  0.029      0.449    0.385E-19  0.029
0.4115   0.608E-19  0.029      0.450    0.608E-19  0.029
0.412    0.486E-19  0.029      0.451    0.109E-18  0.029
0.413    0.831E-19  0.029      0.4515   0.932E-19  0.029
0.4135   0.648E-19  0.029      0.452    0.122E-18  0.029
0.414    0.750E-19  0.029      0.453    0.239E-18  0.029
0.4145   0.811E-19  0.029      0.454    0.170E-18  0.029
0.415    0.811E-19  0.029      0.455    0.340E-18  0.029
0.4155   0.689E-19  0.029      0.4555   0.405E-18  0.029
0.416    0.426E-19  0.029      0.456    0.101E-18  0.029
0.417    0.486E-19  0.029      0.457    0.162E-19  0.029
0.418    0.588E-19  0.029      0.458    0.122E-19  0.029
0.419    0.669E-19  0.029      0.4585   0.142E-19  0.029
0.420    0.385E-19  0.029      0.459    0.405E-20  0.029
0.421    0.567E-19  0.029      0.460    0.405E-20  0.029
0.4215   0.446E-19  0.029      0.4605   0.608E-20  0.029
0.422    0.527E-19  0.029      0.461    0.203E-20  0.029
```

0.462 0.000E+00 0.029

Methy Glyoxyl, below 350 NM

```
MeGly,<350nm
! METHYL GLYOXYL ABSORB COEFS, PLUM ET AL, 1983
! Carter assumed the QY for wl<0.350 was 1.00.
Points
! Units (cm^2/molecule) further multiplied by 1000
FAC=1000.0
  0.220  0.210E-20
  0.225  0.210E-20
  0.230  0.421E-20
  0.235  0.757E-20
  0.240  0.925E-20
  0.245  0.841E-20
  0.250  0.925E-20
  0.255  0.925E-20
  0.260  0.967E-20
  0.265  0.105E-19
  0.270  0.126E-19
  0.275  0.143E-19
  0.280  0.151E-19
  0.285  0.143E-19
  0.290  0.147E-19
  0.295  0.118E-19
  0.300  0.114E-19
  0.305  0.925E-20
  0.310  0.631E-20
  0.315  0.547E-20
  0.320  0.336E-20
  0.325  0.168E-20
  0.330  0.841E-21
  0.335  0.000E+00
```

Methyl Glyoxyl, above 350 NM

```
MeGly,>350nm
!METHYL GLYOXYL ABSORB COEFS, PLUM ET AL, 1983
! Plum estimated qy for pyrex filtered EC chamber.
!  CCOCHO + HV = HO2O. + CO + CCO-O2O. + RCO3O.
Points
! Units (cm^2/molecule) further multiplied by 1000
FAC=1000.0
  0.350  0.000E+00  0.107    0.416  0.488E-19  0.107
  0.354  0.421E-21  0.107    0.418  0.505E-19  0.107
  0.358  0.126E-20  0.107    0.420  0.522E-19  0.107
  0.360  0.210E-20  0.107    0.422  0.530E-19  0.107
  0.362  0.210E-20  0.107    0.424  0.517E-19  0.107
  0.364  0.294E-20  0.107    0.426  0.530E-19  0.107
  0.366  0.336E-20  0.107    0.428  0.522E-19  0.107
  0.368  0.421E-20  0.107    0.430  0.555E-19  0.107
  0.370  0.547E-20  0.107    0.432  0.513E-19  0.107
  0.372  0.589E-20  0.107    0.434  0.568E-19  0.107
  0.374  0.757E-20  0.107    0.436  0.622E-19  0.107
  0.376  0.799E-20  0.107    0.438  0.606E-19  0.107
  0.378  0.883E-20  0.107    0.440  0.547E-19  0.107
  0.380  0.101E-19  0.107    0.441  0.614E-19  0.107
  0.382  0.109E-19  0.107    0.442  0.547E-19  0.107
  0.384  0.135E-19  0.107    0.443  0.555E-19  0.107
  0.386  0.151E-19  0.107    0.4435 0.681E-19  0.107
  0.388  0.172E-19  0.107    0.444  0.597E-19  0.107
  0.390  0.206E-19  0.107    0.445  0.513E-19  0.107
  0.392  0.210E-19  0.107    0.446  0.488E-19  0.107
  0.394  0.231E-19  0.107    0.447  0.572E-19  0.107
  0.396  0.248E-19  0.107    0.448  0.547E-19  0.107
  0.398  0.261E-19  0.107    0.449  0.656E-19  0.107
  0.400  0.278E-19  0.107    0.450  0.505E-19  0.107
  0.402  0.299E-19  0.107    0.451  0.303E-19  0.107
  0.404  0.320E-19  0.107    0.452  0.429E-19  0.107
  0.406  0.379E-19  0.107    0.453  0.278E-19  0.107
  0.408  0.395E-19  0.107    0.454  0.227E-19  0.107
  0.410  0.433E-19  0.107    0.456  0.177E-19  0.107
  0.412  0.471E-19  0.107    0.458  0.841E-20  0.107
  0.414  0.479E-19  0.107    0.460  0.421E-20  0.107
                             0.464  0.168E-20  0.107
```

CAL Aromatic Unknown

```
AROMUNK
! These data supplied by Carter on Floppy 9/87.
! The value is constant for wl <= 350 and is based upon
!  value needed to fit chamber data for aromatics runs.
Points
!
! Units are (cm^2/molecule) multiplied by 1000
FAC=1000.0
0.200     3.95E-20 0.1
0.350     3.95E-20 0.1
0.360     0.0
```

D

Flux Model Code

SOLCALC

Mon 10-10-88 22:52:23 i CONTENTS

		Page	Line
06-06-88 08:52:30			
SOLCALC.PAS	Solar Time and Position Routines	1	1
SOLCALC.PAS	UNC Site Location constants	1	6
SOLCALC.PAS	Global variables	1	14
SOLCALC.PAS	Interface Section	1	28
SOLCALC.PAS	JulDay	1	29
SOLCALC.PAS	SetLocation	1	33
SOLCALC.PAS	SetJulDay	1	36
SOLCALC.PAS	SolarOrbit	1	41
SOLCALC.PAS	SunRiseSetTime	1	43
SOLCALC.PAS	TrueSolarTime	1	46
SOLCALC.PAS	SolarCoords	1	48
SOLCALC.PAS	InitLocalTime	1	50
SOLCALC.PAS	AdvanceTime	1	53
SOLCALC.PAS	local static variables	1	58
SOLCALC.PAS	Implementation Section	2	72
SOLCALC.PAS	JulDay	2	73
SOLCALC.PAS	SetLocation	2	95
SOLCALC.PAS	SetJulDay	2	113
SOLCALC.PAS	SolarOrbit	2	124
SOLCALC.PAS	SunRiseSetTime	4	209
SOLCALC.PAS	TrueSolarTime	4	239
SOLCALC.PAS	SolarCoords	5	250
SOLCALC.PAS	InitLocalTime	5	282
SOLCALC.PAS	AdvanceTime	5	303
SOLCALC.PAS		6	322

```
   06-06-88 08:52:30  SOLCALC.RAS    Solar Time and Position Routines                    Pg   1
Mon 10-10-88 22:52:23                UNC Site Location constants                         of   6
                                                                                        1-62

 1  (<t>Solar Time and Position Routines)
 2  UNIT SolCalc;
 3  INTERFACE
 4   Uses crt,trigfunc,mathfunc;
 5
 6  (<s>UNC Site Location constants)
 7   Const
 8    SiteLat       = 35.7083;(deg)
 9    SiteLong      = 79.1085;(deg)
10    SiteTZ        = 4; ( EDT )
11
12   Type float = single;
13
14  (<s>Global variables)
15   Var
16    SolDec,   (solar declination, radians)
17    SolRadius,(earth-sun relative radius )
18    SolZen,   (solar zenith angle, radians)
19    SolEle,   (solar elevation, radians)
20    SolAzm,   (solar azmiuth angle, radians)
21    LocalTimeDiff, (hh.ff)
22    SolEqt    (equation of time, hh.ff)
23     : float;
24
25    SolCosZA, (cos solar zenith angle)
26    SolSinZA  (sin solar zenith angle)
27     : extended;
28  (<s>Interface Section)
29    Function JulDay(month,day,year:float):extended;
30        (JD=0 = Jan 1, 4713 BC = (-4712 Jan 1) at GreenWich Mean Noon = 12h UT )
31
32    (Usage: if UNC site, use constants above as call parameter values)
33    Procedure SetLocation(LocLat(deg),LocLong(deg),LocTZ(hh.ff): float);
34
35    (USAGE: Sets internal variable D to the Julian Day number from Jan. 1, 1900 at Noon)
36    Procedure SetJulDay(MM,DD,YY: float);
37
38    (USAGE:   expects D to be Julian Days since Jan.1, 1900,
39              sets global variables (eg SolDec, SolRadius, SolEqt)
40              sets internal variables (eg sindec, cosdec ))
41    Procedure SolarOrbit;
42
43    Procedure SunRiseSetTime(VAR HalfDayLength, TimeTwilight,
44                                TimeSunRise, SNoon, TimeSunSet: float);
45
46    Function TrueSolarTime(CT(clock time, hh.ff): float):float;
47
48    Procedure SolarCoords(CT (clock time, hh.ff): float);
49
50    Procedure InitLocalTime(Var LocalTime, StartingTime: float;
51                            TimeSunRise, increment: float);
52
53    Function AdvanceTime(Var LocalTime: float; EndingTime,
54                         TimeSunSet, increment: float): boolean;
55
56  IMPLEMENTATION (******** Solar Time and Position Routines ************)
57
58  (<s>local static variables)
59  VAR
60   Lambda_in_hrs,
61   sindec,   (sin and cos of declination)
62   cosdec,
```

```
  sinlat,   (sin and cos of latitude)
  coslat,
  D         ( number Julian days since Jan. 1, 1900 )
    : extended;

  HaveLocation,
  HaveTime,
  HaveOrbit : boolean;

(<s>Implementation Section)
 Function JulDay(month,day,year:float):extended;
  var
   y1, y2, m1, d1 : extended; ( needed to avoid 8087 stack overflow )
  Begin
   ( Function is valid for 1801 to 2099 )
   if NOT ((1801 <= year) and (year <= 2099))
    then
     begin
      Writeln('Error--year out of range in JulDay.');
      halt;
     end;
   y1 := 367*year;
   y2 := int(7*(year+int((month+9)/12))/4);
   m1 := int(275*month/9) + day;
   d1 := 1721013.5 - 0.5*sign(100*year+month - 190002.5)) + 0.5;
   JulDay := y1-y2+m1+d1;
       ( Ref: Almanac for Computers, 1984, Nautical Almanac Office,
         United States Naval Observatory
         Washington DC, 20390 )
  End;

(Usage: if UNC site, use constants above as parameter values in call)
Procedure SetLocation(LocLat(deg.ff),LocLong(deg.ff),LocTZ(hh.ff): float);
 var
  Latitude,
  Longitude,
  StdTimeZone : float;
 begin
  Latitude    := LocLat*rpd;
  Longitude   := LocLong*rpd;
  StdTimeZone := LocTZ;
  sinlat      := sin(Latitude);
  coslat      := cos(Latitude);
  Lambda_in_hrs := 12*Longitude/Pi;
  LocalTimeDiff := StdTimeZone - Lambda_in_hrs;
  HaveLocation := true;
 end;

(USAGE: returns the Julian Day number from Jan. 1, 1900 at Noon.
        assumes that yy is 19yy. )
Procedure SetJulDay(MM,DD,YY: float);
 var
  DDo : float;
 begin
  DDo := DD + (Lambda_in_hrs+12)/24; ( noon on this day )
  D := JulDay(MM,DDo,YY) - JulDay(1,0,1900);
  HaveTime := true;
 end;

(USAGE:   expects D to be Julian Days since Jan.1, 1900,
          sets local global variables (eg SolDec, SolEqt))
Procedure SolarOrbit;
```

```
125     VAR
126      T,             { Universal Time since Jan 1, 1900 in Julian Years of
127                                        365.25 days, scaled by 100 }
128      M0,            { mean anomaly }
129      ecc,           { eccentricity }
130      E,             { eccentric anomaly }
131      V,             { true anomaly }
132      eps,           { obliquity }
133      omeg,          { long of perigee }
134      L1,            { luna longitude }
135      NutLong,       { longitude of Nutation }
136      meanLong,
137      Y,
138      alpha0,        { solar longitude }
139      alpha          { Solar Right Ascention, rads }
140      {a0,}          { sra in, hours }
141        : extended;
142      i : integer;
143     {Global to this function are variables:
144      dec,       solar declination
145      sindec,
146      cosdec,
147      eqt        equation of time
148
149      HaveLocation,
150      HaveTime,
151      HaveOrbit      coordination flags
152
153      Formulas are from Barkstrom, Bruce, "What Time Does the Sun
154              Rise and Set?", Byte, Vol 6, No. 7 p 94, July 1981
155     }
156     Begin { SolarOrbit }
157      if not (HaveLocation and HaveTime)
158       then
159        begin
160         writeln('No location or time before calling SolarOrbit: Stopping.');
161         halt;
162        end;
163      T := D/36525.0;
164      ecc := 0.01675104 - (4.08E-5 + 1.26E-7 * T) * T;
165      { Compute the MeanAnomaly }
166      M0  := -1.52417 + (1.50E-4 + 3.0E-6 * T) * sqr(T) +
167               0.9856002670 * D;
168      if M0 > 360 then M0 := M0 - 360 * int(M0/360);
169      M0 := M0*rpd;
170      E   := M0;
171      For i := 1 to 3 DO
172        E := E + (M0 - (E - ecc * sin(E)))/(1 - ecc * cos(E));
173      V    := Normalize(2 * arctan(sqrt((1+ ecc)/(1-ecc)) * tan(0.5*E)));
174      SolRadius := (1-ecc * cos(E));
175      eps  := (23.452294 - (1.30125E-2 + (1.64E-6 -
176                    5.03E-7 * T) * T) * T) *rpd;
177      omeg := (281.22083 + (4.53E-4 + 3.0E-6 * T)* sqr(T) +
178                    4.70684E-5 * D) *rpd;
179      omeg := Normalize(omeg);
180      { compute Nutation terms --- "wobble caused by moon" }
181      L1 := (259.183275 + (2.078E-3 + 2.0E-6 * T) * sqr(T) -
182                 0.052953922 * D)*rpd;
183      eps := eps + (2.558333333E-3 + 2.5E-7 * T) * cos(L1)*rpd;
184      NutLong := -(4.787222222E-3 + 4.7222222222E-6 * T) * sin(L1)*rpd;
185      { solar declination }
186      sindec := sin(eps) * sin(V+omeg);
```

```
187      cosdec := sqrt(1-sqr(sindec));
188      SolDec := arctan(sindec/cosdec);
189     { eqt of time }
190      meanLong := Normalize(omeg + M0);
191      y := sqr(tan(0.5*eps));
192      y := (1-y)/(1+y);
193      alpha0 := Normalize(omeg + V + NutLong);
194      alpha := arctan(y*tan(alpha0));
195      SolEqt := alpha - meanLong;
196      SolEqt := SolEqt - Pi * int(SolEqt/Pi);
197      if abs(SolEqt) > 0.9*Pi
198        then
199         if SolEqt > 0
200          then
201            SolEqt := SolEqt - Pi
202          else
203            SolEqt := SolEqt + Pi;
204     { a0  := 12 * Normalize(eqt + meanLong)/Pi ; }
205      SolEqt := -12 * SolEqt/Pi; { convert to hours.ff }
206      HaveOrbit := true;
207    end; { Solar Orbit }
208
209   Procedure SunRiseSetTime(VAR HalfDayLength, TimeTwilight, TimeSunRise,
210                                SNoon, TimeSunSet: float);
211    Const
212     cosZSR = -0.01454;  { z = 90d50s center of sun when upper limb visible }
213     cosZTL = -0.10453;  { z = 96d    center of sun when civilian twilight  }
214    { TST == True Solar Time
215      ClkT== Std Clock Time
216      M   == observers meridian  (79.1 deg for UNC site)
217      Mo  == std meridian (75 deg for EST)
218      Local Time Diff = 4(min/deg)(M-Mo) + time zone corrections (EST to EDT)
219      MST == Mean Solar Time = ClkT + LocalTimeDiff
220      TST  = (MST + SolEqt)
221           := ClkT + LocalTimeDiff + SolEqt
222      ClkT := TST  - LocalTimeDiff - SolEqt
223    }
224    Begin
225     if not HaveOrbit
226      then
227       begin
228        writeln('No Sunrise because Solar Orbit not calculated yet:Stopping.');
229        halt;
230       end;
231     HalfDayLength:= 12*arccos((cosZTL - sindec*sinlat)/(cosdec*coslat))/pi;
232     TimeTwilight := 12.0(TST) - HalfDayLength - LocalTimeDiff - SolEqt;
233     HalfDayLength:= 12*arccos((cosZSR - sindec*sinlat)/(cosdec*coslat))/pi;
234     TimeSunSet   := 12.0(TST) + HalfDayLength  - LocalTimeDiff - SolEqt;
235     SNoon        := 12.0(TST) - LocalTimeDiff  - SolEqt;
236     TimeSunRise  := 12.0(TST) - HalfDayLength  - LocalTimeDiff - SolEqt;
237    end; { SunRiseSetTime }
238
239   Function TrueSolarTime(CT(clock time, hh.ff):float):float;
240    begin
241     if not HaveOrbit
242      then
243       begin
244        writeln('Solar Orbit not calculated yet:Stopping.');
245        halt;
246       end;
247     TrueSolarTime := CT + LocalTimeDiff + SolEqt;
248    end;
```

```
249
250   Procedure SolarCoords(CT {clock time, hh.ff}:float);
251    var
252     hrangle,coshrangle,sinele,cosazm: extended;
253    begin
254     if not HaveOrbit
255      then
256       begin
257        writeln('Solar Orbit not calculated yet:Stopping.');
258        halt;
259       end;
260     { hrangle is positive in the mornings }
261     hrangle := (12.0 - TrueSolarTime(CT)) * 15*rpd;
262     coshrangle := cos(hrangle);
263     { see p15, Iqbal, Solar Radiation -- and cos za = sin ele }
264     sinele := (sindec*sinlat+cosdec*coslat*coshrangle);
265     if (sinele> 1.000001) then cosazm:= 1.0000000;
266     SolCosZA := sinele;
267     SolZen := arccos(SolCosZA);
268     SolSinZA := sin(SolZen);
269     SolEle := arcsin(sinele);
270     cosazm := ((sinele*sinlat-sindec)/(cos(SolEle)*coslat));
271     if (cosazm> 1.0000000001) then cosazm:= 0.99999999999999999;
272     if (cosazm<-1.0000000001) then cosazm:=-0.99999999999999999;
273     SolAzm := arccos(cosazm);
274     {if (cosazm>=0) then ((0<=SolAzm) and (SolAzm<= halfpi)))
275     {if (cosazm<=0) then ((halfpi<=SolAzm) and (SolAzm<=pi)))
276
277     if hrangle < 0
278      then SolAzm := -SolAzm;
279     { SolAzm is pos in the east and neg in the west }
280    end;
281
282   Procedure InitLocalTime(Var LocalTime, StartingTime: float;
283                                TimeSunRise, increment: float);
284    begin
285     { advance at least 5 minutes past sunrise }
286     if StartingTime < TimeSunRise
287      then
288       begin
289        LocalTime := TimeSunRise + 5/60.0;
290        while (StartingTime < LocalTime) do
291          StartingTime := StartingTime + increment;
292        StartingTime := StartingTime - increment;
293        { StartingTime <= TimeSunRise <= StartingTime+increment)
294        { LocalTime == TimeSunRise + 5 mins }
295       end
296      else
297       LocalTime := StartingTime;
298     { reset Julian day to local time and recalc declination }
299     D := D - 0.5 {noon} + (LocalTime-TimeSunRise)/24.0;
300     SolarOrbit;
301    end;
302
303   Function AdvanceTime(Var LocalTime: float; EndingTime,
304                              TimeSunSet, increment: float): boolean;
305    begin
306     if (LocalTime < TimeSunSet) and (LocalTime < EndingTime)
307      then
308       begin
309        LocalTime := LocalTime + increment;
310        D := D + (increment/24.0);
```

```
311         SolarOrbit;
312         AdvanceTime := true;
313        end
314       else
315         AdvanceTime := false;
316     end;
317
318   begin
319    HaveLocation:= false;
320    HaveTime     := false;
321    HaveOrbit    := false;
322   end.
```

$

AEROMOD

Mon 10-10-88 22:44:40 i CONTENTS

		Page	Line
05-29-88 18:17:14			
AEROMOD.PAS		1	1
AEROMOD.PAS	Aerosol Model Routines Used in Broadband and Spect	1	2
AEROMOD.PAS	Physical Conditions Variables	1	26
AEROMOD.PAS	Aerosol Model Variables	1	44
AEROMOD.PAS	Init_Physical	2	77
AEROMOD.PAS	Adj_Physical	2	78
AEROMOD.PAS	Init_Aerosol	2	80
AEROMOD.PAS	Adj_Aerosol	2	81
AEROMOD.PAS	Fz	2	84
AEROMOD.PAS	Init_Physical	2	97
AEROMOD.PAS	Adj_Physical	2	104
AEROMOD.PAS	Init_Aerosol	3	136
AEROMOD.PAS	Adj_Aerosol	3	169
AEROMOD.PAS		5	265

```
Unit AeroMod;
{<t> Aerosol Model Routines Used in Broadband and Spectral Models }
{            A Component of JSPECTRA Model.                       }
{        Copyright, 1987,1988, (c) H. Jeffries                    }
{                    May 27, 1988                                 }
{Usage Notes:
 o calling routine should set a value of z, the zenith angle in DEGREES,
   and assign a value to mu0, the cos before calling Init_Physical,
   and Init_Aerosol.
 o calling routine should set a value in frh, the fractional rel humidity,
   before calling Adj_Aerosol.
}

INTERFACE
 Uses
  TrigFunc,MathFunc;

Type
 float     =single;
 Amodel    =(User, SFrural, SFurban);

Const
 am_ref = 1.73; {relative air mass for reflection calculation}

Var {share these with bb and spectral units}
{<s> Physical Conditions Variables}
 z,       {zenith angle in degrees}
 mu0,     {cos of zenith angle}
 sqmu0,   {cos^2 zenith angle}
 cubmu0,  {cos^3 zenith angle}
 formu0,  {cos^4 zenith angle}
 pres,    {atmospheric pressure}
 o3,      {total column ozone (atm cm)}
 w,       {total column precipitable water vapor (atm cm)}
 frh,     {fractional rel humidity}
 am,      {relative Rayleigh air mass}
 amp,     {presure corrected air mass}
 amau,    {upper aerosol air mass}
 amo,     {ozone air mass}
 amw,     {water air mass}
 gamma    {Rayleigh diffuse transmission exponent as fn of amp}
  : float;

{<s>Aerosol Model Variables}
 AerosolModel : Amodel; {type of model selected}
 oda,     {total optical depth for aerosol extinction}
 odas,    {total optical depth for aerosol scattering}
 odaa,    {total optical depth for aerosol absorption}
 odau,    {upper aerosol optical depth}
 odasu,   {optical depth for upper aerosol scattering}
 odaau,   {optical depth for upper aerosol absorption}
 odal,    {lower aerosol optical average depth, dry aerosol}
 oda50,   {aerosol optical depth at 0.50 microns}
 oda38,   {aerosol optical depth at 0.38 microns (for bb model)}
 odabb,   {aerosol optical depth for total broadband model}
 odauv,   {aerosol optical depth for Eppley UV bb model}

 alpha,   {aerosol wavelength variation factor}
 alpha1,  {aerosol wavelength variation factor < 0.5 microns}
 alpha2,  {aerosol wavelength variation factor > 0.5 microns}
 gg,      {aerosol asymmetry factor}
 alg, a_fs, b_fs, {terms in forward scattering fraction}
```

```
63    fs,      {fraction aerosol scatter forward}
64    fsp,     {fraction aerosol scatter forward at am_ref}
65    af_a,    {actinic anisotropic factor}
66    af_s,    {actinic aerosol forward scatter factor}
67    omega,   {single scattering aerosol albedo for lower aerosol}
68    omegau,  {single scattering aerosol albedo for upper aerosol}
69    omega_x,{user specified single scattering aerosol alebdo for lower aerosol}
70    omega_l,{omega at wavelength omega_w}
71    omega_p,{wavelength variation factor}
72    omega_w,{base wavelength for omega_l}
73    omega_bb,{single scattering aerosol albedo at 0.5 for total bb model}
74    omega_uv {single scattering aerosol albedo in uv for Epply bb model}
75     : float;
76
77  Procedure Init_Physical(Upres:float; Uo3,Uw:float);
78  Procedure Adj_Physical;
79
80  Procedure Init_Aerosol(Uodal,Ufrha,Uodau,Uomega:float; UAeroModel: integer);
81  Procedure Adj_Aerosol;
82
83  IMPLEMENTATION
84  Function Fz(mu,h:float): float;
85   {calc slant length relative to zenith from altitude h in km}
86   {mu = cos zenith angle}
87   {formula is from Robinson, Solar Radiation}
88   const
89     R = 6370; {radius of earth in km}
90   var
91    rh : float;
92   begin
93    rh := R/h;
94    Fz:=sqrt(sqr(rh*mu)+2*rh+1)-rh*mu;
95   end;
96
97  Procedure Init_Physical(Upres,Uo3,Uw: float);
98   begin
99    pres   := Upres;
100   o3     := Uo3;
101   w      := Uw;
102  end;
103
104 Procedure Adj_Physical;
105  begin
106   sqmu0  := mu0*mu0;
107   cubmu0 := sqmu0*mu0;
108   formu0 := sqmu0*sqmu0;
109
110   {relative optical air mass for this zenith angle.}
111   am  := 1.0/(mu0+0.15*pow((93.885-z),-1.253));
112   {limit the air mass to less than 24 because of accuracy of some
113    formulas used later in which am is raised to a power}
114   if (am > 24) then am:=24;
115   {pressure corrected air mass.}
116   amp := am*pres/1013.0;
117
118   {Rayleigh diffuse transmission modifier for this zenith angle}
119   gamma := 0.85*( (pow(amp,1.019)-amp) + exp(-0.48*(pow(amp,1.35)-amp)) );
120
121   {ozone air mass for this zenith angle.}
122   {height of o3 layer is 20km to 40km}
123   amo := (40*Fz(mu0,40)-20*Fz(mu0,20))/(40-20);
124
```

```
125    {water vapor air mass for this zenith angle.}
126    amw := 1.0/(mu0+0.0548*pow((92.650-z),-1.452));
127    {limit the water vapor air mass to less than 24 because of accuracy of some
128     formulas used later in which am is raised to a power}
129    if (amw > 24) then amw:=24;
130
131    {upper aerosol layer air mass for this zenith angle.}
132    {height of upper aerosol layer is 15km to 25km}
133    amau:= (25*Fz(mu0,25)-15*Fz(mu0,15))/(25-15);
134   end;
135
136  Procedure Init_Aerosol(Uodal,Ufrha,Uodau,Uomega:float; UAeroModel: integer);
137   begin
138    AerosolModel := Amodel(UAeroModel);
139    {adj average aerosol od to be dry (0%RH) od}
140    odal    := Uodal/(1/(1-0.84*pow(Ufrha,3.8)));
141    omega_x:= Uomega;
142
143    {upper aerosol model -- only odau is changed, rest is default}
144    odau    := Uodau;
145    omegau := 0.942; {use default rural value}
146    odasu    := omegau*odau;
147    odaau    := (1-omegau)*odau;
148
149    {set parameters for lower aerosol model}
150    case AerosolModel of
151     SFrural:
152      begin
153       omega_p := 0.05; {wavelength variation factor}
154       omega_w := 0.4;
155      end;
156     SFurban:
157      begin
158       omega_p := 0.03;
159       omega_w := 0.55;
160      end;
161     User:
162      begin
163       omega_p := 0.04;
164       omega_w := 0.5;
165      end;
166    end;
167   end; {Init_Aerosol}
168
169  Procedure Adj_Aerosol;
170   begin
171    Case AerosolModel of
172     SFrural:
173      begin
174       {adjust aerosol parameters for rel humidity}
175       if frh < 0.7
176        then
177         begin
178          alpha1  := 0.8;
179          alpha2  := 1.3;
180          omega_l := 0.9462;
181          gg      := 0.66;
182         end
183        else {0.7 <= frh <= 0.99}
184         begin { adjust omega and alpha for humidity }
185          if (frh <= 0.95)
186           then alpha1 := 0.8
```

```
187           else alpha1 := 0.7;
188          if (frh < 0.99)
189           then alpha2 := 1.505/(1+0.605*pow(frh,3.8))
190           else alpha2 := 0.90;
191          omega_l := 0.9462 + 0.1393*(frh-0.7);
192          gg      := 0.6638 + 0.3721*(frh-0.7);
193         end; {humidity adjustments}
194        end; {SFrural }
195      SFurban:
196       begin
197        {adjust aerosol parameters for rel humidity}
198        if frh < 0.7
199         then
200          begin
201           alpha1  := 0.8;
202           alpha2  := 1.3;
203           omega_l := 0.88;
204           gg      := 0.64;
205          end
206         else {0.7 <= frh <= 0.99}
207          begin { adjust omega and alpha for humidity }
208           if (frh <= 0.95)
209            then alpha1 := 0.8
210            else alpha1 := 0.7;
211          if (frh < 0.99)
212           then alpha2 := 1.505/(1+0.605*pow(frh,3.8))
213           else alpha2 := 0.90;
214          omega_l := 0.88 + 0.1393*(frh-0.7);
215          gg      := 0.64 + 0.3721*(frh-0.7);
216         end; {humidity adjustments}
217        end; {SFurban }
218      User:
219       begin
220        {adjust aerosol parameters for rel humidity}
221        if frh < 0.7
222         then
223          begin
224           alpha1  := 0.8;
225           omega_l := omega_x;
226           gg      := 0.64;
227          end
228         else {0.7 <= frh <= 0.99}
229          begin { adjust omega and alpha for humidity }
230           if (frh <= 0.95)
231            then alpha1 := 0.8
232            else alpha1 := 0.7;
233          if (frh < 0.99)
234           then alpha2 := 1.505/(1+0.605*pow(frh,3.8))
235           else alpha2 := 0.90;
236          omega_l := omega_x + 0.1393*(frh-0.7);
237          gg      := 0.64    + 0.3721*(frh-0.7);
238         end; {humidity adjustments}
239      end; {User }
240    end; {case}
241    alg:=ln(1.0-gg);
242    a_fs:=alg*(1.459+alg*(0.1595+alg*0.4129));
243    b_fs:=alg*(0.0783+alg*(-0.3824-alg*0.5874));
244    fs     := 1.0-0.5*exp((a_fs+b_fs*mu0)*mu0);
245    fsp    := 1.0-0.5*exp((a_fs+b_fs/am_ref)/am_ref);
246
247    af_a := (0.384112+6.283352*sqmu0-7.83516*cubmu0+3.073598*formu0)*
248              exp(pow(amp,1.004)-pow(amp,0.975));
```

```
249     af_s := 1.05-0.35190*sqmu0+0.188733*cubmu0;
250
251     (adjust od50 for rel humidity effect on dry oda1)
252     oda50:= oda1*(1/(1-0.84*pow(frh,3.8)));
253
254     (calculate optical densities for broadband models)
255     omega_bb := omega_1*exp(-omega_p*sqr(ln(0.50/omega_w)));
256     omega_uv := omega_1*exp(-omega_p*sqr(ln(0.38/omega_w)));
257     oda38:= oda50*pow((0.38/0.5),-alpha1); (UV od)
258     odauv:= 0.8200*oda38;                  (weighted aero od for UV)
259     odabb:= 0.2758*oda38+0.35*oda50;       (weighted aero od for total)
260   end;
261
262 begin (unit initilization)
263   z  :=0.0;
264   mu0:=1.0;
265 end.
$
```

BBMODEL

Mon 10-10-88 22:45:24 i CONTENTS

		Page	Line
06-01-88 10:37:40			
BBMODEL.PAS		1	1
BBMODEL.PAS	Total Solar Radiation and UltraViolet Broadband Mo	1	2
BBMODEL.PAS	Irradiance Components Variables	1	21
BBMODEL.PAS	InitBBConditions	1	37
BBMODEL.PAS	Broadband	1	40
BBMODEL.PAS	Transmissions	1	45
BBMODEL.PAS	InitBBConditions	2	83
BBMODEL.PAS	Broadband	2	108
BBMODEL.PAS		4	244

```
Unit BBMODEL;
(<t> Total Solar Radiation and UltraViolet Broadband Model        )
(            A Component of JSPECTRA Model.                       )
(        Copyright, 1987,1988, (c) H. Jeffries                    )
(                    May 27, 1988                                )
INTERFACE
 Uses
  AeroMod,TrigFunc,MathFunc
{$IFDEF BBDebug}
  ,DebugIO
{$ENDIF}
 ;

Type
 float     =single;
const
  tCvtCal   = 1.4335E-3;{cal/m2/min per watt/m2}
  uCvtCal   = 1.4335;   {millical/m2/min per watt/m2}

Var {share these with main programs}
 {<s>Irradiance Components Variables}
 bbE_g,  {broadband global irradiance}
 bbE_n,  {broadband normal irradiance}
 bbE_d,  {broadband direct irradiance}
 bbE_dif,{broadband diffuse}
 bbE_rgd,{broadband reflection}
 brhoa,  {broadband air reflectivity}

 uvE_g,  {Eppley UV global irradiance}
 uvE_n,  {Eppley UV normal irradiance}
 uvE_d,  {Eppley UV direct irradiance}
 uvE_dif,{Eppley UV diffuse}
 uvE_rgd,{Eppley UV gnd-sky reflection}
 urhoa   {Eppley UV broadband air reflectivity}
   : float;

Procedure InitBBConditions(ESdist,Upres,Uo3,Uw,Uodal,Ufrha,Uodau,Uomega:float;
                           UAeroModel: integer);

Procedure Broadband(Ufrh, ZA{zenith angle, radians): float;
                    Var ModTsr, ModUV{cal/cm^2/min): float);

IMPLEMENTATION
Var {private variables}
 {<s>Transmissions}
 btr,    {broadband transmittance func for molecular rayleigh scattering}
 Uo,Uw,  {terms in ozone transmission}
 baov,   {absorption by ozone in visible}
 baou,   {basorption by ozone in UV}
 bto3,   {broadband transmittance for ozone absorption}
 btg,    {broadband transmittance for uniformly mixed gas absorption}
 btw,    {broadband transmittance for water vapor absorption}
 btasu,  {broadband transmittance for upper aerosol scattering}
 btaau,  {broadband transmittance for upper aerosol absorption}
 btal,   {broadband transmittance for lower aerosol extinction}
 btasl,  {broadband transmittance for lower aerosol scattering}
 btaal,  {broadband transmittance for lower aerosol absorption}
 btas,   {broadband lower and upper transmittance aerosol scattering}
 btaa,   {broadband lower and upper total transmittance aerosol absorption}
 bt_D,   {broadband total normal transmission}
 bt_S,   {broadband total diffuse transmission}
 bt_M,   {broadband total reflection transmission}
```

```
63   utr,    {UVe transmittance for molecular rayleigh scattering}
64   uto3,   {UVe transmittance for ozone absorption}
65   utal,   {UVe transmittance for lower aerosol extinction}
66   utasl,  {UVe transmittance for lower aerosol scattering}
67   utaal,  {UVe transmittance for lower aerosol absorption}
68   utas,   {UVe lower and upper transmittance aerosol scattering}
69   utaa,   {UVe lower and upper total transmittance aerosol absorption}
70   ut_D,   {UVe total normal transmission}
71   ut_S,   {UVe total diffuse transmission}
72   ut_M,   {UVe total reflection transmission}
73   rho,    {broadband ground reflectivity, eg, albedo}
74
75   uvRelRB,{UVe Relative Response as function of za for direct beam}
76   uvRelRS,{UVe Relative Response as function of za for diffuse rad}
77
78   bbET,   {broadband extraterrestrial irradiance==solar constant}
79   uvET    {UVe extraterrestrial UV irradiance 300--385nm}
80
81    : float;
82
83  Procedure InitBBConditions(ESdist,Upres,Uo3,Uw,Uodal,Ufrha,Uodau,Uomega:float;
84                             UAeroModel: integer);
85  Begin
86   {Extraterrestrial flux at earth-sun rel distance}
87   bbET   := (1366.97{watt/m2)-15.55(@298nm)-31.99(@2800+nm))/sqr(ESdist);
88   uvET   := (93.11(@385nm)-16.54(@300nm))/sqr(ESdist);
89
90   Init_Physical(Upres,Uo3,Uw);
91   Init_Aerosol(Uodal,Ufrha,Uodau,Uomega,UAeroModel);
92
93  {$IFDEF BBdebug}
94    if DebugON
95     then
96      begin
97       write(DBfile,'"mu0"  "amp"  "gamma" ');
98       write(DBfile, '"alg" "a_fs"  "b_fs" "fs"');
99       write(DBfile, '"btr"  "bto3"  "(btw*btg)" "btas"  "btaa" ');
100      write(DBfile, '"utr"  "uto3"  "utas" "utaa" ');
101      write(DBfile, '"bt_D" "ut_D" ');
102      write(DBfile, '"bt_S"  "ut_S" "bbE_d"  "uvE_d" "bbE_dif"  "uvE_dif" ');
103      writeln(DBfile, '"bbE_g"  "uvE_g"');
104     end;
105 {$ENDIF}
106  end;
107
108 Procedure Broadband(Ufrh, ZA{zenith angle, radians): float;
109                     Var ModTsr, ModUV: float);
110  begin
111   if (Ufrh < 1.0)
112    then AeroMod.frh := Ufrh
113    else AeroMod.frh := 0.999;
114   if (0 <= ZA) and (ZA < (pi/2)))
115    then
116     begin
117      AeroMod.mu0 := cos(ZA);
118      AeroMod.z   := ZA/rpd;  {z=zenith angle in degrees}
119     end
120    else
121     begin
122      ModTsr:=0.0;
123      ModUV :=0.0;
124      exit;
```

```
125      end;
126
127    Adj_Physical;
128    Adj_Aerosol;
129
130    Uo := o3*amp;
131    Uw := w*amw;
132
133    (broadband transmittances)
134    btr   := exp(-0.085*pow(amp,0.84)(*(1+amp-pow(amp,1.01)))); (0.0903=Bird value)
135             (note that values of amp>29 cause floating overflow in orig formula)
136    baov  := 0.00240*Uo/(1+0.044*Uo+3E-4*sqr(Uo)); (0.002715==Bird value)
137    baou  := 0.14*Uo/pow((1+139.48*Uo),0.3035);  (0.1611==Bird value)
138    bto3  := 1.0-baov-baou;
139    btw   := 1.0-2.4959*Uw/(pow((1+79.034*Uw),0.6828)+6.385*Uw); (Bird)
140    btg   := exp(-0.0127*pow(amp,0.26));
141    btasu := exp(-odasu*amau);
142    btaau := exp(-odaau*amau);
143    btal  := exp( -pow(odabb,0.873)*(1+odabb-pow(odabb,0.7088))*pow(amp,0.9108) );
144    btaal := 1-(1-omega_bb)*(1-amp+pow(amp,1.06))*(1-btal);
145    btasl := btal/btaal;
146    btas  := btasu*btasl;
147    btaa  := btaau*btaal;
148
149    (Eppley Ultraviolet broadband transmissions)
150    utr   := exp(-0.63*pow(amp,0.81)(*(1+amp-pow(amp,1.008))) ); (0.66=fitted value)
151    uto3  := exp(-0.35*o3*amo*(1-amo+pow(amo,1.1)));
152    utal  := exp( -pow(odauv,0.873)*(1+odauv-pow(odauv,0.7088))*pow(amp,0.9108) );
153    utaal := 1-(1-omega_uv)*(1-amp+pow(amp,1.03))*(1-utal);
154    utasl := utal/utaal;
155    utas  := btasu*utasl; (upper aerosol bb and uv od are same)
156    utaa  := btaau*utaal;
157
158    (total direct transmission)
159    bt_D := btr*bto3*btw*btg*btaa*btas;
160    ut_D := utr*uto3*utaa*utas;
161    (normal and direct component of irradiance onto surface)
162    bbE_n := bbET*bt_D;
163    bbE_d := bbE_n*mu0;
164    uvE_n := uvET*ut_D;
165
166    (The Eppley Sensor has a 13% cos response error for direct beam.
167     The uvRelRB response model below is based on data in
168     Drummond and Wade paper on the design of the Eppley sensor.
169     We also know that the Eppley Sensor has a cos response error for
170     the diffuse radiation as well, but do not have adequate measurements
171     to fully characterize its response.  The uvRelRS response model below
172     is based upon fitting the morning reading.)
173     if (mu0>0.3420)
174      then
175       begin
176        uvRelRB := 1.000 - 0.1302*(1.0000-mu0);
177        uvRelRS := 1.000;
178       end
179      else
180       begin
181        uvRelRB := 0.915 + 0.1222*(0.3420-mu0);
182        uvRelRS := 1.000 + 0.5000*(0.3420-mu0);
183       end;
184
185    (Paper says Eppley normalized at 45 deg, where uvRelRB = 0.962,
186     but we find the correction fits the data best at a
```

```
187     lower normalized value of 0.92.)
188    uvE_d := (uvRelRB/0.92)*uvE_n*mu0;
189
190    (diffuse component of irradiance onto surface)
191    bt_S    := 0.79*bto3*btw*btg*btaa*
192               ((1.0-btr)*0.5 + (1.0-btas)*fs)/(1-amp+pow(amp,1.02));
193    bbE_dif := bbET*mu0*bt_S;
194
195    ut_S    := 0.79*uto3*utaa*
196               ((1.0-pow(utr,gamma))*0.5 + (1.0-utas)*fs)/(1-amp+pow(amp,1.05));
197    uvE_dif := (uvRelRS/0.92)*uvET*mu0*ut_S;
198
199    (total broadband albedo is assumed to be constant)
200    rho := 0.20;
201
202    (atmospheric reflectivity for average air mass)
203    brhoa:= (0.0685+(1-fs)*(1-btas));
204
205    (albedo contribution to diffuse component)
206    bt_M    := ((bt_D+bt_S)*rho*brhoa)/(1.0-rho*brhoa);
207    bbE_rgd := bbET*mu0*bt_M;
208
209    (uv bb albedo is assumed to be constant)
210    rho := 0.05;
211
212    (atmospheric uv reflectivity for average air mass)
213    urhoa:= (0.385+(1-fs)*(1-utas));
214
215    (albedo contribution to diffuse component (corrected for Eppley Response))
216    ut_M    := ((ut_D+ut_S)*rho*urhoa)/(1.0-rho*urhoa);
217    uvE_rgd := (uvRelRS/0.92)*uvET*mu0*ut_M;
218
219    (total on a horizontal surface)
220    bbE_g := bbE_d+bbE_dif+bbE_rgd;
221    uvE_g := uvE_d+uvE_dif+uvE_rgd;
222
223    (convert watt/m^2 to cal/cm^2/min)
224    ModTsr:= bbE_g*tCvtCal;
225    ModUV := uvE_g*uCvtCal;
226
227  ($IFDEF BBDebug)
228    if DebugON
229     then
230      begin
231       write(DBfile,  mu0:5:3,' ',amp:5:2,' ',gamma:5:2,' ');
232       write(DBfile,  alg:5:3,' ',a_fs:5:3,' ',b_fs:5:3,' ',fs:5:3,' ');
233       write(DBfile,  btr:5:3,' ',bto3:5:3,' ',(btw*btg):5:3,' ',btas:5:3,' ',btaa:5:3,' ');
234       write(DBfile,  utr:5:3,' ',uto3:5:3,' ',utas:5:3,' ',utaa:5:3,' ');
235       write(DBfile, bt_D:5:3,' ',ut_D:5:3,' ');
236       write(DBfile, bt_S:5:3,' ',ut_S:5:3,' ',bbE_d:5:3,' ',uvE_d:5:3,' ',
237                                               bbE_dif:5:3,' ',uvE_dif:5:3,' ');
238       writeln(DBfile,bbE_g:6:2,' ',uvE_g:5:3);
239      end;
240  ($ENDIF)
241   end (BroadBand);
242
243  begin (unit initialization)
244  end.
```

JSMODEL

Mon 10-10-88 22:48:41 i CONTENTS

		Page	Line
06-21-88 23:49:44			
JSMODEL.PAS		1	1
JSMODEL.PAS	Spectral Solar Radiation Model	1	2
JSMODEL.PAS	"Irradiances"	1	22
JSMODEL.PAS	InitJSConditions	1	34
JSMODEL.PAS	AdjJSConditions	1	37
JSMODEL.PAS	SpectralWave	1	39
JSMODEL.PAS	SpectralFlux	1	41
JSMODEL.PAS	Extraterrestrial solar flux and coefficients	1	51
JSMODEL.PAS	"Optical Densities and transmissions"	1	60
JSMODEL.PAS	ReadETSpect	2	93
JSMODEL.PAS	Init_Albedo	2	124
JSMODEL.PAS	Albedo	3	137
JSMODEL.PAS	InitJSConditions	3	148
JSMODEL.PAS	AdjJSConditions	3	181
JSMODEL.PAS	SpectralWave	4	202
JSMODEL.PAS	SpectralFlux	4	207
JSMODEL.PAS		5	294

```
Unit JSMODEL;
{<t>          Spectral Solar Radiation Model                    }
{             A Component of JSPECTRA Model.                    }
{         Copyright, 1987,1988, (c) H. Jeffries                 }
{                    May 27, 1988                               }
INTERFACE
 Uses
  AeroMod,MathFunc,TrigFunc,FileOpen,RunPath
{$IFDEF JSDebug}
  ,DebugIO
{$ENDIF}
 ;

Type
 float     =single;
const
 {output units are photons/cm^2/min/nm}
 Cvtflux = 3.02042169E+13{=(1/hc)*((60sec/min)/1E-4m2/cm2)};
 MaxNumWv = 280;

Var {share these with main programs}
{<s>"Irradiances"}
 E_g,     {global on horizontal surface}
 E_n,     {direct normal irradiance}
 E_d,     {direct on surface}
 E_dif,   {diffuse irradiance}
 E_rgd,   {multiple reflection between ground & air component of diffuse}
 aE_g,    {global actinic flux}
 aE_dif,  {diffuse actinic flux}
 aE_rgd   {multiple reflection actinic flux}
  :float;
 NumWv : integer; {number of ET flux wavelengths read}

Procedure InitJSConditions(ESdist,Upres,Uo3,Uw,Uodal,Ufrha,Uodau,Uomega:float;
                           UAeroModel: integer);

Procedure AdjJSConditions(Ufrh, ZA{zenith angle, radians): float);

Procedure SpectralWave(i: integer; Var lam: float);

Procedure SpectralFlux(i: integer);

IMPLEMENTATION
const
 max_num_albedo=8;

type
 paramvalues = array[0..1,1..max_num_albedo] of float;

Var {private}
{<s>Extraterrestrial solar flux and coefficients}
 wl,      {wavelength, presently in microns}
 h,       {extraterrestrial solar irradiance}
 awa,     {water absorption coeffs}
 aoa,     {ozone abosrption coeffs}
 aua      {uniform gases absorption coeffs}
  : array[1..maxnumwv] of float;
 OldES_corr : float;

{<s>"Optical Densities and transmissions"}
 odr,     {optical depth for molecular rayleigh scattering}
 odo,     {optical depth for ozone absorption}
```

```
63
64   tr,      (transmittance for molecular rayleigh scattering)
65   to3,     (transmittance for ozone absorption)
66   tg,      (transmittance for uniformly mixed gas absorption)
67   tw,      (transmittance for water vapor absorption)
68   tasu,    (transmittance for upper aerosol scattering)
69   taau,    (transmittance for upper aerosol absorption)
70   tasl,    (transmittance for lower aerosol scattering)
71   taal,    (transmittance for lower aerosol absorption)
72   tas,     (total transmittance aerosol scattering)
73   taa,     (total transmittance aerosol absorption)
74   trp,     (tr evaluated at m=1.8)
75   twp,     (tw evaluated at m=1.8)
76   tgp,     (..likewise..)
77   tasp,    (..likewise..)
78   taap,    (..and aerosol too)
79   rho,     (ground reflectivity)
80   rhoa,    (average sky reflectivity for am_ref)
81   t_D,     (direct normal transmission)
82   t_S,     (diffuse irradiance transmission)
83   t_M,     (multiple reflection transmission)
84   at_S,    (diffuse actinic transmission)
85   at_M,    (multiple reflection actinic transmission)
86   view,    (fraction of sky visible to flat surface)
87   aview    (fraction of sky visible to spherical surface)
88    : float;
89  (ground reflectivity at specific wavelengths, eg albedo )
90  ALBE: paramvalues;
91  num_alb, AlbPtr : integer;
92
93 Procedure ReadETSpect;
94  type
95   string80 = string[80];
96  var
97   i          : integer;
98   filename   : filenamestring;
99   EtSpecFile : Text;
100  EBuf       : Array[1..1024] of char;
101  Header     : string80;
102 Begin
103  filename := PgmRunPath + 'ETSPECT.HET';
104  if NOT SetUpFile(1,filename,EtSpecFile,'R',EBuf,SizeOf(EBuf))
105   then
106    begin
107     writeln('Can not find ET Spectrum File '+filename);
108     halt;
109    end;
110  repeat
111   readln(EtSpecFile, Header);
112  until ((Header[1]<>'!') and (length(Header) > 0));
113  i := 1;
114  while (not eof(EtSpecFile)) and (i <= maxnumwv) do
115   begin
116    readln(EtSpecFile, wl[i], h[i], awa[i], aoa[i], aua[i]);
117    i := succ(i);
118   end;
119  NumWv := i-1;
120  OldES_corr:=1.0; {ET file is at Earth-Sun dist of 1.0}
121  Close(EtSpecFile);
122 end;
123
124 Procedure Init_Albedo;
```

```
125   Begin
126    num_alb := 8;
127    ALBE[0,1] := 0.28;  ALBE[1,1] := 0.05;
128    ALBE[0,2] := 0.38;  ALBE[1,2] := 0.05;
129    ALBE[0,3] := 0.40;  ALBE[1,3] := 0.06;
130    ALBE[0,4] := 0.43;  ALBE[1,4] := 0.08;
131    ALBE[0,5] := 0.50;  ALBE[1,5] := 0.10;
132    ALBE[0,6] := 0.70;  ALBE[1,6] := 0.08;
133    ALBE[0,7] := 0.78;  ALBE[1,7] := 0.25;
134    ALBE[0,8] := 4.00;  ALBE[1,8] := 0.25;
135  end;
136
137  Function Albedo(lam:float):float;
138   var
139    u,rho : float;
140   begin
141    While (lam >= ALBE[0,AlbPtr+1]) DO AlbPtr := succ(AlbPtr);
142    u := (lam-ALBE[0,AlbPtr])/(ALBE[0,AlbPtr+1]-ALBE[0,AlbPtr]);
143    rho := (1-u)*ALBE[1,AlbPtr] + u*ALBE[1,AlbPtr+1];
144    if rho < 0.0 then rho := 0.0;
145    Albedo:=rho;
146   end;
147
148  Procedure InitJSConditions(ESdist,Upres,Uo3,Uw,Uodal,Ufrha,Uodau,Uomega:float;
149                             UAeroModel: integer);
150   var
151    ES_corr : float;
152    i : integer;
153   Begin
154    (correct ET flux for Earth Sun dist)
155    ES_corr := 1.0/sqr(ESdist);
156    for i:= 1 to NumWv do
157     h[i]:=h[i]*ES_corr/OldES_corr;
158    OldES_corr:=ES_corr; (save change for next time)
159    AlbPtr := 1;
160    view   := 1.0;
161    aview  := 1.0;
162
163    (Init_Albedo; -- done in startup code.)
164    Init_Physical(Upres,Uo3,Uw);
165    Init_Aerosol(Uodal,Ufrha,Uodau,Uomega,UAeroModel);
166
167  ($IFDEF JSdebug)
168     if DebugON
169      then
170       begin
171        write(DBfile,'"lam"  "mu0"  "amp"  "gamma" ');
172        write(DBfile, '"alg" "a_fs"  "b_fs" "fs"');
173        write(DBfile, '"tr"  "to3"  "(tw*tg)" "tas"  "taa" ');
174        write(DBfile, '"t_D" "t_S" "at_S"  "rho" "rhoa" ');
175        write(DBfile, '"E_n"  "E_dif"  "aE_dif" ');
176        writeln(DBfile, '"E_g"  "aE_g"');
177       end;
178  ($ENDIF)
179   end;
180
181  Procedure AdjJSConditions(Ufrh, ZA(zenith angle, radians): float);
182   begin
183    if (Ufrh < 1.0)
184     then AeroMod.frh := Ufrh
185     else AeroMod.frh := 0.999;
186    if ((0 <= ZA) and (ZA < (pi/2)))
```

```
187     then
188      begin
189       AeroMod.mu0 := cos(ZA);
190       AeroMod.z   := ZA/rpd;  (z=zenith angle in degrees)
191      end
192     else
193      begin
194       AeroMod.mu0:=0.0;
195       AeroMod.z  :=90.0;
196      end;
197
198    Adj_Physical;
199    Adj_Aerosol;
200   end;
201
202  Procedure SpectralWave(i: integer; Var lam: float);
203   begin
204    lam := wl[i];
205   end;
206
207  Procedure SpectralFlux(i: integer);
208   var
209    lam : float;
210   begin
211    lam := wl[i];
212
213    (calculate optical depths for this wavelength)
214    odr := 1.0/( sqr(lam)*(115.6406*sqr(lam)-1.335) );
215    odo := aoa[i]*o3;
216    (single scatter albedo for lower aerosol)
217    omega := omega_l*exp(-omega_p*sqr(ln(lam/omega_w)));
218    if lam <= 0.5 (microns)
219     then alpha := alpha1
220     else alpha := alpha2;
221    oda  := oda50*pow((lam/0.5),-alpha); (total extinction)
222    odas := omega*oda;                   (scattering component)
223    odaa := (1-omega)*oda;               (absorption component)
224
225    (transmittances)
226    tr   := exp(-odr*amp);
227    to3  := exp(-odo*amo);
228    tw   := exp(-0.2385*awa[i]*w*amw/pow((1.0+20.07*awa[i]*w*amw),0.45));
229    tg   := exp(-1.41*aua[i]*amp/(pow((1.0+118.3*aua[i]*amp),0.45)));
230    tasu := exp(-odasu*amau);
231    taau := exp(-odaau*amau);
232    tasl := exp(-odas*amp);
233    taal := exp(-odaa*amp);
234    tas  := tasu*tasl;
235    taa  := taau*taal;
236
237    (total direct transmission )
238    t_D := tr*to3*tw*tg*taa*tas;
239    (normal and direct component of irradiance)
240    E_n:=h[i]*t_D;
241    E_d:=E_n*mu0;
242    (diffuse component of irradiance onto surface)
243    t_S   := view*pow(to3,1.1)*tw*tg*taa*
244             ( (1.0-pow(tr,gamma))*0.5 + tr*(1.0-tas)*fs );
245    E_dif := h[i]*mu0*t_S;
246
247    (diffuse component of actinic flux)
248    at_S  := aview*af_a*pow(to3,1.3)*tw*tg*taa*
```

```
249             ( tas*(1.0-tr)*0.5 + tr*(1.0-tas)*af_s );
250     aE_dif:= h[i]*at_S;
251
252     (atmospheric reflectivity for average air mass of am_ref)
253     trp    := exp(-odr*am_ref);
254     twp    := exp(-0.2385*awa[i]*w*am_ref/(pow((1.0+20.07*awa[i]*w*am_ref),0.45)));
255     tgp    := exp(-1.41*aua[i]*am_ref/(pow((1.0+118.3*aua[i]*am_ref),0.45)));
256     tasp   := exp(-odas*am_ref);
257     taap   := exp(-odaa*am_ref);
258     rhoa   := view*tgp*twp*taap*
259                 (0.5*(1.0-pow(trp,2.0))*fsp+(1.0-fsp)*(1.0-tasp)*trp);
260
261     (albedo contribution to diffuse component)
262     rho    := Albedo(lam);
263     t_M    := ((t_D+t_S)*rho*rhoa)/(1.0-rho*rhoa);
264     E_rgd := h[i]*mu0*t_M;
265
266     (total on a horizontal surface)
267     E_g := E_d+E_dif+E_rgd;
268
269     (albedo contribution to actinic flux)
270     at_M   := 2*mu0*(t_D+t_S)*(rho+rho*rhoa)/(1.0-rho*rhoa);
271     aE_rgd := h[i]*at_M;
272
273     (total actinic flux)
274     aE_g := E_n+aE_dif+aE_rgd;
275
276   ($IFDEF JSDebug)
277     if DebugON
278      then
279       begin
280        write(DBfile,  lam:5:3,' ',mu0:5:3,' ',amp:5:2,' ',gamma:5:2,' ');
281        write(DBfile,  alg:5:3,' ',a_fs:5:3,' ',b_fs:5:3,' ',fs:5:3,' ');
282        write(DBfile,  tr:5:3,' ',to3:5:3,' ',(tw*tg):5:3,' ',tas:5:3,' ',taa:5:3,' ');
283        write(DBfile,  t_D:5:3,' ',t_D:5:3,' ',rho:5:3,' ',rhoa:5:3,' ');
284        write(DBfile,  t_S:5:3,' ',at_S:5:3,' ',E_n:5:3,' ',
285                                                E_dif:5:3,' ',aE_dif:5:3,' ');
286        writeln(DBfile,E_g:6:2,' ',aE_g:5:3);
287       end;
288   ($ENDIF)
289    end; (SpectralFlux)
290
291   begin (unit initialization)
292    ReadETSpect;
293    Init_Albedo;
294   end.
```

$

CHMMODEL

Mon 10-10-88 22:47:31 i CONTENTS

		Page	Line
06-25-88 09:56:02			
CHMMODEL.PAS		1	1
CHMMODEL.PAS	Chamber Transmissions As Function of Sun's Positio	1	2
CHMMODEL.PAS	ChamTrans	1	11
CHMMODEL.PAS	ChamFloor	1	23
CHMMODEL.PAS	InitChamTrans	1	28
CHMMODEL.PAS	InternalInit	3	131
CHMMODEL.PAS	InitChamTrans	3	143
CHMMODEL.PAS	ThruRedEnd	3	160
CHMMODEL.PAS	The 5-mil Teflon Film Transmission model.	4	210
CHMMODEL.PAS	FilmTrans	4	211
CHMMODEL.PAS	ChamTrans	4	231
CHMMODEL.PAS	ChamFloor	6	338
CHMMODEL.PAS		6	351

06-25-88 09:56:02 CHMMODEL.PAS Chamber Transmissions As Function of Sun's Positio Pg 1
Mon 10-10-88 22:47:31 ChamTrans of 6
1-62

```
1  UNIT ChmModel;
2  {<t> Chamber Transmissions As Function of Sun's Position        }
3  {    Chamber Floor Reflection Characteristics                   }
4  {             Last Modified June 25, 1988                      }
5  INTERFACE
6  Uses
7   TrigFunc, MathFunc, Proj3D, HuntTab, HuntFind;
8  Type
9   float = single;
10
11 Procedure ChamTrans(SolZA{radians}, SolAzm{radians}: float;
12                      VAR BTrans90,BTrans35,AAlpha,DTrans90,Ref,
13                          ETrans35, EDTrans35: float);
14  {BTrans90  is chamber area weighted beam trans at 900 nm}
15  {BTrans35  is chamber area weighted beam trans at 350 nm}
16  {AAlpha    is beam trans wavelength variation factor in:
17     Trans_lam = BTrans90-exp(-AAlpha*(lam-110)) }
18  {DTrans90  is chamber average diffuse transmission at 900 nm }
19  {Ref       is chamber total weighted reflection at 350 and 550 nm}
20  {ETrans35  is beam transmission at in-chamber Eppely UV Radiometer location.}
21  {EDTrans35 is diff transmission at in-chamber Eppely UV Radiometer location.}
22
23 Procedure ChamFloor(lam:float; Var rho, rhorho: float);
24  {lam       is wavelength in nanometers}
25  {rho       is chamber floor reflectivity at lam}
26  {rhorho    is chamber back reflection at lam}
27
28 Procedure InitChamTrans;
29  {called before each day's use of ChamTran}
30
31 IMPLEMENTATION
32 const
33  {----------------- Chamber Characteristics ---------------}
34  { Chamber face azimuth }
35  BlueEndAzm    = +153 {deg from S};
36  FieldSideAzm  =   63 {deg from S};
37  RedEndAzm     =  -27 {deg from S};
38  LabSideAzm    = -117 {deg from S};
39  { Chamber face tilt }
40  BlueEndTilt   =  90.0 {deg from horizontal};
41  FieldSideTilt =  52.7 {deg from horizontal};
42  RedEndTilt    =  90.0 {deg from horizontal};
43  LabSideTilt   =  52.7 {deg from horizontal};
44  { Chamber face area }
45  BlueEndArea   =   23.50 {m^2};
46  FieldSideArea =  102.84 {m^2};
47  RedEndArea    =   23.50 {m^2};
48  LabSideArea   =  102.84 {m^2};
49
50  { Red End Panel corner co-ordinates (x,y,z) in feet from center of chamber}
51  BottField : array[1..3] of float = (  3.52, -23.85,  3.33);
52  BottLab   : array[1..3] of float = (-21.64, -10.64,  3.33);
53  TopLab    : array[1..3] of float = ( -9.41, -17.06, 21.50);
54  TopField  : array[1..3] of float = ( -8.71, -17.43, 21.50);
55  { In-chamber Eppley UV radiometer location in feet from center of chamber }
56  EUV : array[1..3] of float = (-1.71, -11.26, 3.33);
57
58 {Chamber floor reflection and chamber back reflection}
59  Wavelength: array[1..10] of float =
60   ( 300,  380,  400,  430,  500,  550,  700,  780,  850,  2000);
61  rho_floor : array[1..10] of float =
62   (0.570,0.593,0.599,0.607,0.624,0.674,0.698,0.686,0.686,0.700);
```

```
63   rho_walls : array[1..10] of float =
64    (0.048,0.060,0.072,0.096,0.100,0.100,0.101,0.101,0.102,0.120);
65
66  (Simple Model of Diffuse Transmission)
67   numdifZA=10;
68   numdifAZ=15;
69   (x=columns == zenith angles)
70   difZA: array[1..numdifZA] of float=(0,10,20,30,40,50,60,70,80,90);
71   (y=rows == azmuith angles)
72   difAZ: array[1..numdifAZ] of float=(150,90,75,60,45,30,15,0,
73                                       -15,-30,-45,-60,-75,-90,-150);
74   (matrix is diff transmission at 480nm with y=rows(azmuith) and x=cols(zenith) )
75   AvgDifTran: array[1..numdifAZ,1..numdifZA] of float=(
76    (0.78,0.78,0.78,0.78,0.82,0.84,0.84,0.83,0.78,0.78),
77    (0.78,0.78,0.78,0.78,0.82,0.84,0.84,0.83,0.78,0.78),
78    (0.78,0.78,0.78,0.78,0.83,0.84,0.84,0.83,0.78,0.78),
79    (0.78,0.78,0.78,0.78,0.84,0.84,0.84,0.82,0.78,0.78),
80    (0.78,0.78,0.78,0.78,0.84,0.84,0.84,0.82,0.78,0.78),
81    (0.78,0.78,0.78,0.78,0.82,0.84,0.84,0.83,0.78,0.78),
82    (0.78,0.78,0.78,0.78,0.80,0.83,0.83,0.82,0.78,0.78),
83    (0.78,0.78,0.78,0.78,0.78,0.78,0.80,0.80,0.78,0.78),
84    (0.78,0.78,0.78,0.78,0.74,0.74,0.74,0.74,0.78,0.78),
85    (0.78,0.78,0.78,0.78,0.74,0.71,0.70,0.70,0.78,0.78),
86    (0.78,0.78,0.78,0.78,0.74,0.72,0.70,0.70,0.78,0.78),
87    (0.78,0.78,0.78,0.78,0.76,0.76,0.78,0.78,0.78,0.78),
88    (0.78,0.78,0.78,0.78,0.78,0.80,0.82,0.82,0.78,0.78),
89    (0.78,0.78,0.78,0.78,0.82,0.83,0.84,0.84,0.78,0.78),
90    (0.78,0.78,0.78,0.78,0.82,0.83,0.84,0.84,0.78,0.78));
91
92  VAR { private }
93   sinBlueEndTilt,      (sin of chamber side tilt angle)
94   sinFieldSideTilt,
95   sinRedEndTilt,
96   sinLabSideTilt,
97   cosBlueEndTilt,      (cos of chamber side tilt angle)
98   cosFieldSideTilt,
99   cosRedEndTilt,
100  cosLabSideTilt,
101  cosBlueEndZA,        (cos of sun's angle on film)
102  cosFieldSideZA,
103  cosRedEndZA,
104  cosLabSideZA,
105  effBlueEndArea,      (projected area of sides)
106  effFieldSideArea,
107  effRedEndArea,
108  effLabSideArea,
109  wtTransBlueEnd,      (fractional contribution of side to total area)
110  wtTransFieldSide,
111  wtTransRedEnd,
112  wtTransLabSide,
113  wtRefBlueEnd,        (fractional of total projected area for interior sides)
114  wtRefFieldSide,
115  wtRefRedEnd,
116  wtRefLabSide,
117  TransBlueEnd,        (350 and 550 nm side transmissions)
118  TransFieldSide,
119  TransRedEnd,
120  TransLabSide,
121  RefBlueEnd,          (350 and 550 nm side internal reflections)
122  RefFieldSide,
123  RefRedEnd,
124  RefLabSide,
```

```
125   TotEffArea
126      :float;
127
128   jAZ,kZA,iCF : integer;
129   BeenInside: boolean;
130
131  Procedure InternalInit;
132   Begin
133    sinFieldSideTilt := sin(FieldSideTilt*rpd);
134    sinRedEndTilt    := sin(RedEndTilt*rpd);
135    sinLabSideTilt   := sin(LabSideTilt*rpd);
136    sinBlueEndTilt   := sin(BlueEndTilt*rpd);
137    cosFieldSideTilt := cos(FieldSideTilt*rpd);
138    cosRedEndTilt    := cos(RedEndTilt*rpd);
139    cosLabSideTilt   := cos(LabSideTilt*rpd);
140    cosBlueEndTilt   := cos(BlueEndTilt*rpd);
141   end;
142
143  Procedure InitChamTrans;
144   begin
145    jAZ:=1;
146    kZA:=1;
147    iCF:=1;
148    BeenInside:=false;
149   end;
150
151  {The In-chamber Eppley Radiometer sees the sun through particular
152   Teflon panels at particular times and thus a specific transmission
153   calculation that accounts for the view geometry is needed in contrast
154   to the area weighted average for the whole chamber volume.  A major
155   factor for the Eppley UV sensor is whether the direct beam is comming
156   through the FieldSide film or through the RedEnd Panel film.  The
157   following function uses projective geometry to determine this for
158   a given solar zenith and azimuth angle.}
159
160  Function ThruRedEnd(ZA,Azm:float):boolean;
161   const
162    numpt = 4;
163   var
164    i,i1: integer;
165    x,y,z: array[1..numpt] of float;
166    sine  : float;
167    Inside: boolean;
168   begin
169    {Method is to project the Red End panel triangle (actually a trapozoid)
170     on to a plane using as a viewpoint the in-chamber Eppley UV sensor
171     from the sun's position. Then test if the projected x,y coord of the
172     Eppley sensor (which is always 0,0) is inside the polygon formed by the
173     projected x,y coord of the Red End panel. The inside test is from
174     "IUCALC-A", R.G. Edwards, etal, Oak Ridge, ORNL/CSD/TM-12, August 1976.
175     Because the projected Eppley sensor co-ord is always 0,0, it is not
176     calculated explicitly. The SetView and Transform routines are in the
177     Proj3D unit.  The bigD factor of 3500 and mag of 1 only affects the
178     z values which are not used in this calculation anyway.  Points
179     go in clockwise order looking at outside End from sun.}
180
181    SetView(((pi/2)-ZA),Azm,3500,EUV[1],EUV[2],EUV[3],1);
182    Transform(BottField[1], BottField[2], BottField[3], x[1],y[1],z[1]);
183    Transform(BottLab[1],   BottLab[2],   BottLab[3],   x[2],y[2],z[2]);
184    Transform(TopLab[1],    TopLab[2],    TopLab[3],    x[3],y[3],z[3]);
185    Transform(TopField[1],  TopField[2],  TopField[3],  x[4],y[4],z[4]);
186
```

```
187    Inside:=false;
188    for i:= 1 to numpt do
189     begin
190      i1:=i+1;
191      if (i1>numpt) then i1:=1;
192      sine:=( (0-x[i])*(y[i1]-y[i])-(0-y[i])*(x[i1]-x[i]) );
193      if (sine = 0)
194       then
195        begin
196         if ( ((0-x[i])*(x[i1]-x[i])+(0-y[i])*(y[i1]-y[i]))*
197              ((0-x[i1])*(x[i1]-x[i])+(0-y[i1])*(y[i1]-y[i])) > 0)
198          then ThruRedEnd:=true
199          else ThruRedEnd:=false;
200         exit;
201        end
202       else
203        if ( ( (0<=y[i1]) and (0>=y[i]) and (sine > 0) ) or
204             ( (0> y[i1]) and (0<=y[i]) and (sine < 0) ) )
205          then Inside := NOT Inside;
206     end;
207    ThruRedEnd:=Inside;
208   end;
209
210  (<s>The 5-mil Teflon Film Transmission model.)
211  Procedure FilmTrans(lam, mu :float; VAR tlam, rlam :float);
212   (lam  is the wavelength at which the trans is desired)
213   (mu   is the cos of the angle between the film normal and the solar beam)
214   (tlam is the calculated beam transmission)
215   (rlam is the calculated beam reflection)
216   const
217    t900_0 = 0.895;
218    alpha_0= 0.008563;
219    sqrgamma = 0.8010 (= (t900_0)^2);
220    lam0  = 110 (nm);
221   var
222    alpha, t900, tlam_0 : float;
223   begin
224    tlam_0 := t900_0 - exp(-alpha_0*(lam-lam0));
225    t900 := sqrt(2*sqrgamma*mu - sqrgamma*mu*mu);
226    alpha:= 0.010538-0.00685*mu+0.004875*mu*mu;
227    tlam := t900-exp(-alpha*(lam-lam0));
228    rlam := tlam_0-tlam;
229   end;
230
231  Procedure ChamTrans(SolZA,SolAzm(radians): float;
232                  VAR BTrans90,BTrans35,AAlpha,DTrans90,Ref,
233                  ETrans35, EDTrans35: float);
234   var
235    SolCosZA,SolSinZA: float;
236   begin
237    SolSinZA := sin(SolZA);
238    SolCosZA := cos(SolZA);
239    cosBlueEndZA := cosBlueEndTilt*SolCosZA +
240                    sinBlueEndTilt*SolSinZA*cos(SolAzm-BlueEndAzm*rpd);
241    cosFieldSideZA := cosFieldSideTilt*SolCosZA +
242                    sinFieldSideTilt*SolSinZA*cos(SolAzm-FieldSideAzm*rpd);
243    cosRedEndZA := cosRedEndTilt*SolCosZA +
244                    sinRedEndTilt*SolSinZA*cos(SolAzm-RedEndAzm*rpd);
245    cosLabSideZA := cosLabSideTilt*SolCosZA +
246                    sinLabSideTilt*SolSinZA*cos(SolAzm-LabSideAzm*rpd);
247
248    effBlueEndArea   :=     BlueEndArea   * cosBlueEndZA;
```

```
249    effFieldSideArea := 0.5*FieldSideArea * cosFieldSideZA;
250    effRedEndArea    :=     RedEndArea    * cosRedEndZA;
251    effLabSideArea   := 0.5*LabSideArea   * cosLabSideZA;
252
253    TotEffArea := ord(effBlueEndArea   > 0)*effBlueEndArea   +
254                  ord(effFieldSideArea > 0)*effFieldSideArea +
255                  ord(effRedEndArea    > 0)*effRedEndArea    +
256                  ord(effLabSideArea   > 0)*effLabSideArea;
257
258    wtTransBlueEnd   := ord(effBlueEndArea   > 0)*effBlueEndArea/TotEffArea;
259    wtTransFieldSide := ord(effFieldSideArea > 0)*effFieldSideArea/TotEffArea;
260    wtTransRedEnd    := ord(effRedEndArea    > 0)*effRedEndArea/TotEffArea;
261    wtTransLabSide   := ord(effLabSideArea   > 0)*effLabSideArea/TotEffArea;
262
263    wtRefBlueEnd     := ord(effBlueEndArea   < 0)*-effBlueEndArea/TotEffArea;
264    wtRefFieldSide   := ord(effFieldSideArea < 0)*-effFieldSideArea/TotEffArea;
265    wtRefRedEnd      := ord(effRedEndArea    < 0)*-effRedEndArea/TotEffArea;
266    wtRefLabSide     := ord(effLabSideArea   < 0)*-effLabSideArea/TotEffArea;
267
268    (Calculate the film transmission at the max response wavelength
269     of 350 nm for the Eppley Sensor)
270    FilmTrans(350,abs(cosBlueEndZA),TransBlueEnd,RefBlueEnd);
271    FilmTrans(350,abs(cosFieldSideZA),TransFieldSide,RefFieldSide);
272    FilmTrans(350,abs(cosRedEndZA),TransRedEnd,RefRedEnd);
273    FilmTrans(350,abs(cosLabSideZA),TransLabSide,RefLabSide);
274
275    BTrans35 := wtTransBlueEnd*TransBlueEnd + wtTransFieldSide*TransFieldSide +
276                wtTransRedEnd*TransRedEnd + wtTransLabSide*TransLabSide;
277
278    Ref   := wtRefBlueEnd*RefBlueEnd*(1 + wtTransFieldSide*TransFieldSide) +
279             wtRefFieldSide*RefFieldSide +
280             wtRefRedEnd*RefRedEnd   + wtRefLabSide*RefLabSide;
281
282    (If sun shining on outside of Red End panel and can see
283      Eppley sensor thru the End then use Red End trans else
284      use the Field or Lab side transmission.)
285    (The vertical Red End Panel is "more" diffusing than the
286      side panels (based upon direct observations) and its
287      beam transmission is less, while its diffuse trans is higher.)
288    (The side panel film has a slight "droop" that leads to a slightly
289      more normal incident of sun on part of film, resulting in a larger
290      beam trans than is calculated assuming a perfectly flat panel.)
291    if (cosRedEndZA >0) and ThruRedEnd(SolZA,SolAzm)
292      then
293       begin
294        ETrans35 := 0.96*TransRedEnd;
295        EDTrans35:= 1.04*TransRedEnd;
296        BeenInside:=True;
297       end
298      else
299       begin
300        if Not BeenInside
301         then
302          begin
303           ETrans35 := 1.03*TransFieldSide;
304           EDTrans35:= 1.02*TransFieldSide;
305          end
306         else
307          begin
308           ETrans35 := TransLabSide;
309           EDTrans35:= TransLabSide;
310          end;
```

```
311       end;
312
313    {Calculate the film trans at the ref wavelength, 900 nm}
314    FilmTrans(900,abs(cosBlueEndZA),TransBlueEnd,RefBlueEnd);
315    FilmTrans(900,abs(cosFieldSideZA),TransFieldSide,RefFieldSide);
316    FilmTrans(900,abs(cosRedEndZA),TransRedEnd,RefRedEnd);
317    FilmTrans(900,abs(cosLabSideZA),TransLabSide,RefLabSide);
318
319    BTrans90 := wtTransBlueEnd*TransBlueEnd + wtTransFieldSide*TransFieldSide +
320             wtTransRedEnd*TransRedEnd + wtTransLabSide*TransLabSide;
321
322    {Calc an area weighted alpha using the 350 and 900 nm trans}
323    AAlpha := -ln(BTrans90-BTrans35)/(350-110);
324
325    {Use "wide-area" average diffuse tranmission table look-up model}
326    if ((SolZA/rpd)<90)
327     then
328      begin
329       DTrans90:=TwoWay(AvgDifTran,difAZ,difZA,numdifAZ,numdifZA,jAZ,kZA,
330                        SolAzm/rpd,SolZA/rpd);
331       {correct AvgDifTran values from 480 to 900 nm }
332       DTrans90:=DTrans90+exp(-AAlpha*(480-110));
333      end
334     else
335      DTrans90:=0.0;
336   end;
337
338  Procedure ChamFloor(lam: float; Var rho,rhorho:float);
339   var
340    u: float;
341   begin
342    if (lam < wavelength[2]) then iCF:=1;
343    while (lam > wavelength[iCF+1]) do inc(iCF);
344    u := (lam-wavelength[iCF])/(wavelength[iCF+1]-wavelength[iCF]);
345    rho    := (1-u)*rho_floor[iCF] + u*rho_floor[iCF+1];
346    rhorho := (1-u)*rho_walls[iCF] + u*rho_walls[iCF+1];
347   end;
348
349  Begin { unit init code }
350   InternalInit;
351  END.
```

$

CHMJSMOD

Mon 10-10-88 22:46:21 i CONTENTS

		Page	Line
08-20-88 14:11:22			
CHMJSMOD.PAS		1	1
CHMJSMOD.PAS	Sig_Error	1	42
CHMJSMOD.PAS	WriteSRWavelength	1	49
CHMJSMOD.PAS	IntegrateFlux	2	80
CHMJSMOD.PAS	ModelLight	2	104
CHMJSMOD.PAS	ShowHelp	3	174
CHMJSMOD.PAS		5	290

```
 1  Program ChmJSModel;
 2  Uses
 3    Crt,Strings,TrigFunc,TimeIO,EscChk,FileOpen,DateUtil,
 4    ChmModel,SolCalc,AeroMod,MetCond,JSmodel;
 5  Const
 6   Title    ='      In Chamber Spectral Spherical Irradiance Model';
 7   Copyright='      Copyright (c), 1988, H.E. Jeffries, K.G. Sexton';
 8   Version  ='                         May 29, 1988';
 9   stop=true;
10   MaxNumInt=92;
11
12  type
13   float=single;
14   wave_array=Array[1..MaxNumWv] of float;
15   spec_array=Array[1..MaxNumWv] of float;
16   datavalue =
17    Record
18     time: float;
19     case t:boolean of
20      true :(lambda : array[1..MaxNumInt] of float);
21      false:(ChmFlux: array[1..MaxNumInt] of float);
22    end;
23  Var
24    jSF : integer; { the SegFile Master list index }
25
26    FComponet : char;
27    HD,twH,srH,NoonH,ssH: float;
28    strdate: string;
29    L,i,mm,dd,yy :integer;
30    Ilam   : array[1..MaxNumInt] of float;
31    lam    : wave_array;
32    Flux   : spec_array;
33
34    Outside: boolean;
35
36    Values : datavalue;
37
38    filepath,filenamein,filenameout,switches: filenamestring;
39    outfile: file of datavalue;
40    Dfile  : text;
41
42  Procedure Sig_Error(mess:string;action:boolean);
43   begin
44    TextColor(red);
45    writeln('ERROR::',mess);
46    if action then halt;
47   end;
48
49  Procedure WriteSRWavelength;
50   var
51    i: integer;
52   begin
53    Values.time:=srH; {first time in file is sunrise}
54    for i:= 1 to NumWv do
55     begin
56      SpectralWave(i,lam[i]);
57      lam[i]:=lam[i]*1000; {convert microns to nanometers}
58     end;
59    with Values do
60     begin
61      For i:=1 to 61 do
62       lambda[i]:= 2.0*i + 298.0;  {last one is 420}
```

```
63      lambda[62]:=430;
64      For i:= 63 to 75 do
65       lambda[i]:=lambda[i-1]+10.0; (last one is 560)
66      lambda[76]:=580.0;
67      For i:= 77 to 85 do
68       lambda[i]:=lambda[i-1]+20.0; (last one is 760)
69      lambda[86]:=800.0;
70      For i:= 87 to 92 do
71       lambda[i]:=lambda[i-1]+40.0; (last one is 1040)
72      (save the values for storage of IFlux below)
73      for i:=1 to MaxNumInt do
74       ilam[i]:=lambda[i];
75     end;
76    write(Outfile,Values); (first values in file are the interval ending wavelengths)
77
78   end;
79
80  Procedure IntegrateFlux;
81   var
82    i,k : integer;
83    IFlux : float;
84   begin
85    with Values do
86     begin
87      IFlux:=2.0*Flux[1];(assume first flux is for 2 nm interval)
88      k:=1;
89      for i:=2 to NumWv do
90       begin
91        if (lam[i]>ilam[k])
92         then
93          begin
94           ChmFlux[k]:=IFlux;
95           inc(k);
96           IFlux:=0.0;
97          end;
98        IFlux:=IFlux+(lam[i]-lam[i-1])*Flux[i];
99       end;
100     ChmFlux[k]:=IFlux; (assign the last one)
101    end;
102   end;
103
104  Procedure ModelLight;
105  var
106    lt : float;
107    ltmin, i : integer;
108    Cloud_T, Cham_dirT, Cham_difT, rho, rhorho,
109    t90,t35,alpha,difT,ref,et35,etd35,
110    OutFlux,
111    ChmE_n, ChmE_d, ChmE_dif, ChmAE_dif, ChmE_ref, ChmE_g, ChmAE_ref,
112     ChmAE_g :float;
113
114   begin (ModelTSRUV)
115    InitChamTrans;
116    InitJSConditions(SolRadius,fpress[jSF],fozone[jSF],fwater[jSF],
117                     fodal[jSF],ffrh10[jSF],fodau[jSF],fomega[jSF],fAmod[jSF]);
118
119    ltmin := round(srH*60);
120    while ((ltmin mod 4) <> 0) do ltmin:=succ(ltmin);
121    lt := (ltmin+8)/60.0;
122    while ((lt < ssH-(8.0/60)) and (lt <= 19.0)) do
123     begin
124      gotoxy(19,15);
```

```
125      write(CvtTimeSHHMM(lt));
126      Values.time := lt; {HH.ff}
127      SolarCoords(lt);
128      AdjJSConditions(dfrh(jSF,lt),SolZen);
129      ChamTrans(SolZen,SolAzm,t90,t35,alpha,difT,ref,et35,etd35);
130      for i:= 1 to NumWv do
131       begin
132        if (lam[i]<400 {nm})
133         then Cloud_T := CloudTrans(jSF,trans_uv,lt)
134         else Cloud_T := CloudTrans(jSF,trans_tsr,lt);
135        SpectralFlux(i);
136        if Outside
137         then
138          begin
139           Case FComponet of
140            'G': OutFlux:=Cloud_T*aE_g;
141            'N': OutFlux:=Cloud_T*E_n;
142            'S': OutFlux:=Cloud_T*aE_dif;
143            'R': OutFlux:=Cloud_T*aE_rgd;
144           end;
145          end
146         else
147          begin
148           Cham_dirT := t90 -exp(-alpha*(lam[i]-110));
149           Cham_difT := difT-exp(-alpha*(lam[i]-110));
150           ChamFloor(lam[i],rho,rhorho);
151           ChmE_n   := Cham_dirT*Cloud_T*E_n;
152           ChmE_d   := mu0*ChmE_n;
153           ChmE_dif := Cham_difT*Cloud_T*(E_dif + E_rgd);
154           ChmAE_dif:= Cham_difT*Cloud_T*(aE_dif+ E_rgd);
155           ChmE_ref := rhorho*(ChmE_d+ChmE_dif)/(1-rhorho);
156           ChmE_g   := ChmE_d+ChmE_dif+ChmE_ref;
157           ChmAE_ref:= 2.0*ChmE_g*(rho+rhorho)/(1-rhorho);
158           ChmAE_g  := ChmE_n+ChmAE_dif+ChmAE_ref;
159           Case FComponet of
160            'G': OutFlux:=ChmAE_g;
161            'N': OutFlux:=ChmE_n;
162            'S': OutFlux:=ChmAE_dif;
163            'R': OutFlux:=ChmAE_ref;
164           end;
165          end;
166        Flux[i]:=(1E-16)*cvtflux*lam[i]*(0.001{nm/micron})*OutFlux;
167       end;
168      IntegrateFlux;
169      write(outfile,Values);
170      lt := lt + 16.0/60.0;
171     end; {lt <= 19}
172   end;
173
174  Procedure ShowHelp;
175   begin
176    writeln(' Expected input:');
177    writeln('   ChmJSMod  metcond.lst[/s/s/s]    OutFilePath'  );
178    writeln('       where /s is an optional switch and one of:');
179    writeln('        /S --- ask for starting File number');
180    writeln('        /O --- produce outside chamber actinic fluxes');
181    writeln('        /Cf--- produce flux component (f=) N,S,R');
182    writeln;
183    writeln('   Output is MMddyy.FIB for inside  binary file and');
184    writeln('             MMddyy.FOB for outside binary file.');
185   end;
186
```

```
187  begin ( main )
188   TextBackground(Black);
189   clrscr;
190   TextColor(Red);
191   writeln(Title);
192   TextColor(Yellow);
193   writeln(Copyright);
194   TextColor(Blue);
195   writeln(Version);
196   TextColor(Green);
197   if (ParamCount = 0)
198    then ShowHelp;
199   TextColor(Cyan);
200   writeln;
201
202   if ParamCount >=1
203    then filenamein := ParamStr(1)
204    else
205     begin
206      write('Enter MetCondition Filename and switches :');
207      readln(filenamein);
208     end;
209   {look for switches in filename}
210   L := pos('/',filenamein);
211   if L <> 0
212    then
213     begin
214      switches := StrUp(copy(filenamein, L, length(filenamein)-L+1));
215      filenamein := copy(filenamein, 1, L-1);
216     end
217    else switches := '';
218
219   writeln('Loading Meterological Conditions file....');
220   LoadMetCond(filenamein);
221
222   if (pos('/O',switches)<>0)
223    then Outside := true
224    else Outside := false;
225
226   if (pos('/S',switches)<>0)
227    then
228     begin
229      write('Enter the index number of the starting File :');
230      readln(jSF);
231      if (jSF>NumSegFiles)
232       then Sig_error('Requested starting File number is too big.',stop);
233     end
234    else
235     jSF := 1;
236
237   i:=pos('/C',switches);
238   if (i<>0)
239    then
240     begin
241      FComponet:=switches[i+2];
242      if (pos(FComponet,'GNSR')=0)
243       then Sig_error('Requested Flux Componet is not one of [GNSR].',stop);
244     end
245    else
246     FComponet:='G';
247
248   if ParamCount >=2
```

```
249    then filepath := ParamStr(2)
250    else
251     begin
252      write('Enter drive and leading path to output FxBFiles :');
253      readln(filepath);
254     end;
255   filepath:=StrUp(filepath);
256
257   TextColor(White);
258   if Escape then;   {clear escape flag}
259   SetLocation(SiteLat,SiteLong,SiteTZ);
260
261   while ((jSF<=NumSegFiles) and (NOT Escape)) do
262    begin
263     filenameout:=copy(fSFdate[jSF],1,pos('.',fSFdate[jSF]));
264     if Outside
265      then filenameout:=filepath+filenameout+'FOB'
266      else filenameout:=filepath+filenameout+'FIB';
267     Assign(Outfile,filenameout);
268     {$I-}Rewrite(Outfile);{$I+}
269     if IOResult<> 0
270      then Sig_Error('Can not find path to FxBFiles.',Stop);
271     Gotoxy(1,14);
272     writeln('Processing file ',filenameout);
273     write(  '          time=');
274     SegFile2Date(fSFdate[jSF],LotusDate,strdate,mm,dd,yy);
275     if yy<70 then yy:=84; {test data set}
276     SetJulDay(mm,dd,(yy+1900));
277     SolarOrbit;
278     SunRiseSetTime(HD,twH,srH,NoonH,ssH);
279     WriteSRWaveLength;
280     ModelLight;
281     close(Outfile);
282     jSF:=succ(jSF); {next SegFile}
283    end; {while jSF<NumSegFiles}
284
285
286   ClrScr;
287   TextColor(Green+Blink);
288   gotoxy(20,18);
289   Writeln(Output,' J S M o d   F I N I S H E D ! ');
290  end.
```

E

Chamber Rate Code

Spectral Integration

Mon 10-10-88 23:13:02 i CONTENTS

		Page	Line
08-14-88 17:24:26			
speclib.pas		1	1
speclib.pas	IntegrateSpec	1	9
speclib.pas	IntegrateSpec	3	75
speclib.pas	ReadASpec	3	95
speclib.pas		4	193

08-14-88 17:24:26 speclib.pas
Mon 10-10-88 23:13:02 IntegrateSpec


```
Unit SpecLib;
(Spectrum Integration Procedure--
 Copyright (c), 1988, H. Jeffries
 Last Modified July 3, 1988 )
INTERFACE
Type
 float=single;

Procedure IntegrateSpec(ReadProc:pointer; Sum: boolean; Class:char;
                        VAR numwl:integer; Var UIWave, UIVal);

 { ReadProc is the address of the procedure that reads the spectrum.

   Sum is either true or false.  If true, the value in the current
   interval is added (summed) to the value in the previous interval
   and thus the current value is the total area from the beginning of
   the integration to the current interval's ending wavelength.
   If false, the current interval value is stored itself and is the
   incremental area between the previous interval's ending wavelength
   and the current interval's ending wavelength.  The former is
   generally used for reaction potential spectra and the latter for
   irradiance spectra.

   Class is from the set:
    'P' points spectrum--line interpolation;
    'p' points spectrum--average bar interpolation;
    'B','C','E' spectral values (ie, per wavelength)
                with beginning, center, ending intervals
    'b','c','e' interval values (ie, *not* per wavelength)
                with beginning, center, ending intervals.
   On call, numwl is the maxsize of UIWave and UIVal array.
   On exit, it is the number of wavelength-val pairs read and is the
   maximum valid location in UIwave and UIVal.

   UIWave and UIVal are the user's wavelength and associated integrated
   or interval spectrum arrays (which are *typeless* to allow various
   sized arrays to be used).  They are overlayed with single dimension
   arrays below. The final units of wl are always nm, but the input file
   can have units of micrometers. ReadProc should convert all input
   units to nanometers if needed.  The final wavelengths for each integral
   or interval value are at the right edge of area panel added.

   'P'oint means a single value located at given wl, and the method is
     not to perform integeration here but later to assume a straight line
     connects the points and to use double trapozoid integration between
     interpolated line values and irradiance intervals.  Thus points are
     the original values stored with the original wavelengths.
   'p'oint means a single value located at a given wl, and the method is
     to use the trapozoid rule to integrate the panel between two points,
     effectively turning the point spectrum into an average bar spectrum.

   Interval means a type of average value over a type of wl interval.
   The types of interval average values in input can be:
    1) units    --- interval averaged value (little letters); or
    2) units/nm --- spectral value (big letters);
   For both of these, the interval width can be variable and
   the input wl coordinate can be given in one of three forms:
    1) the beginning of each interval (B,b)--in which case, the file
       *must* include a final point with a wl-coordinate at the next
       logical interval and have an associated value of 0.0;
    2) the midpoint of each interval, eg, centered (C,c)--the file *must*
       include an initial point with a wl-coordinate less than the first
```

```
63          real centered wl by the first interval width and the value associated
64          with this wl should be 0.0.
65       3) the endpoint of each interval (E,e)--the file *must* include an initial
66          point, with a wl-coordinate at the first logical interval and
67          a value of 0.0.
68      For Interval values, the integration rule is:
69            Add areas and use wavelength at *end* of area panel.  For
70            interval values, the value is the panel area.  For
71            spectral values, the panel area is (value/nm)*interval(nm).
72    }
```

```
73  {<f>}
74  IMPLEMENTATION
75  Procedure IntegrateSpec(ReadProc:pointer; Sum: boolean; Class:char;
76                          VAR numwl:integer; Var UIWave, UIVal);
77   type
78    sarray = array[1..300] of float;
79   var
80    IWave: sarray absolute UIWave; {overlay user's arrays}
81    IVal : sarray absolute UIVal;
82    maxwl: integer; {the size of users IWave and IVal arrays}
83    i, j : integer;
84    wl, val : float;
85    ValOld, IValOld, IValInc, WaveOld, interval: float;
86    PEOF : boolean;
87
88    {This procedure calls whatever procedure the user passes as an address
89     which must have arguments that match the follow order and type. The
90     User's procedure must also be compiled as FAR.  The variables are:
91      Seof -- set true when there are no more wavelength-value pairs
92      w    -- next wavelength
93      v    -- next spectral value associated with wavelength
94     }
95    procedure ReadASpec(Var Seof: boolean; Var w,v: float);
96                        Inline($FF/$9E/ReadProc);
97     {The $FF is a CALL, $9E is a mode byte, 10011110
98      divided as 10 011 110 -- 10  = mem with word disp following
99                               -- 011 = far call
100                              -- 110 = bp+disp
101     ReadProc is the offset relative to the BP reg (see p366.)
102    }
103
104  begin
105   maxwl := numwl;  {On call, numwl is size of users's arrays}
106   numwl := 1;
107   ReadASpec(PEOF,wl,val);
108   {For all Classes, the integrated spectrum before the first (wl,val)
109    in the input file is assumed to be zero.}
110   IValOld := 0.0;
111   IVal[1] := IValOld;
112   WaveOld := wl; {the first wl is assocated with the end of the
113                   integral of all the spectrum up to it, eg 0.0}
114   IWave[1]:= WaveOld;
115   if (Class='p') or (Class='B') or (Class='b')
116    then
117     begin
118      ValOld := val;
119      { For p this becomes the left side of the trapozoidal panel.
120        For B or b this value will be associated with the next wl and
121        the last value in the file should be 0.0; it will not be used.}
122     end;
123   {For 'C','c','E','e' first val should be 0.0 and is ignored anyway}
124   ReadASpec(PEOF,wl,val);
125   while Not PEOF and (numwl < maxwl) do
126    begin
127     inc(numwl);
128     case Class of
129      'P':
130       begin
131        IWave[numwl] := wl;
132        IVal[numwl]  :=val;
133       end;
134      'p':
```

```
135         begin
136          IWave[numwl]:=wl;
137          IValInc:=0.5*(val+ValOld)*(wl-WaveOld);
138          if Sum then IValInc:=IValInc+IValOld;
139          IVal[numwl]:=IvalInc;
140          IvalOld :=IValInc;
141          ValOld  := val;
142          WaveOld :=wl;
143         end;
144        'B','b':
145         begin
146          {previous val used with present wl; present value used next time}
147          IWave[numwl]:=wl;
148          if Class = 'b'
149           then IValInc:=ValOld        {data already multiplied by width}
150           else IValInc:=ValOld*(wl-WaveOld);
151          if Sum then IValInc:=IValInc+IValOld;
152          IVal[numwl] := IValInc;
153          IValOld := IValInc;
154          ValOld  := val;
155          WaveOld := wl;
156         end;
157        'C','c':
158         begin
159          {adj wl used with present val}
160          if (numwl=2)
161           then
162            begin
163             interval:=(wl[ctr]-WaveOld[ctr]); {first point special}
164             IWave[1]:=IWave[1]+interval/2; {adjust 1st to be endpoint}
165            end
166           else
167            {curr}interval:=2.0*((wl[curr]-WaveOld)-{prev}interval/2);
168          IWave[numwl]:=wl[ctr]+{curr}interval/2;
169          if Class = 'c'
170           then IValInc:=val        {data already multiplied by width}
171           else IValInc:=val*interval;
172          if Sum then IValInc:=IValInc+IValOld;
173          IVal[numwl] := IValInc;
174          IValOld := IValInc;
175          WaveOld := wl[ctr];
176         end;
177        'E','e':
178         begin
179          {present value used with present wl}
180          IWave[numwl]:=wl;
181          if Class = 'e'
182           then IValInc:=val        {data already multiplied by width}
183           else IValInc:=val*(wl-WaveOld);
184          if Sum then IValInc:=IValInc+IValOld;
185          IVal[numwl] := IValInc;
186          IValOld:=IValInc;
187          WaveOld := wl;
188         end
189       end; {case}
190       ReadASpec(PEOF,wl,val);
191      end; {while not eof}
192    end;
193  end.
```

CHMRATE

Mon 10-10-88 22:49:46 i CONTENTS

		Page	Line
08-24-88 21:42:36			
CHMRATE.PAS		1	1
CHMRATE.PAS	Sig_Error	3	86
CHMRATE.PAS	ReadCsQy	4	113
CHMRATE.PAS	LoadReactSpec	6	187
CHMRATE.PAS	BarBarRate	7	221
CHMRATE.PAS	BarLineRate	9	287
CHMRATE.PAS	InitVar	11	376
CHMRATE.PAS	CQCmd	11	389
CHMRATE.PAS	CalcRates	12	404
CHMRATE.PAS	SpectralRate	13	454
CHMRATE.PAS	IrrCmd	14	484
CHMRATE.PAS	SingCmd	15	522
CHMRATE.PAS	ShowData	15	533
CHMRATE.PAS		16	609

08-24-88 21:42:36 CHMRATE.PAS
Mon 10-10-88 22:49:46


```
1  Program ChamRate;
2  Uses
3   Crt,       {TP 4.0 fast screen io unit}
4   Dos,       {TP 4.0 DOS services unit}
5   FileOpen,  {Jeffries's standard file open and create unit}
6   ReadEtxt,  {Jeffries's implementation of nested file io}
7   HuntTab,   {Optimized search of linear table}
8   HuntFind,  {Optimized linear interpolation of one-way tables}
9   SpecLib,   {Spectral integration function}
10  Strings,   {useful string operations, eg, string to uppercase}
11  StrIO,     {procedures that allow read to work from strings}
12  TimeIO,    {time conversion for input/output}
13  MathFunc,  {useful math operations, eg pow())}
14  EscChk,    {source of boolean Escape-key function}
15  ErrIO;     {Helps identify errors}
16
17 Const
18  title =     '           Chamber Photolysis Rates By Intergal Average';
19  copyright = '             Copyright, (c) H.E. Jeffries, 1987,1988';
20  version   = '                    Version June 1, 1988';
21
22  maxNumRxn     =  25; {Photolytic reaction cross-section, quantum yields}
23  maxRxnwl      = 150; {CsQy wavelengths}
24  NumSolWl      =  92; {Actinic flux wavelengths}
25
26  stop          = true;
27  showf         = true;
28  noshow        = false;
29
30  ModeSum       = true;
31  ModeInc       = false;
32
33 Type
34  float         = single;
35
36  Rarray        = array[1..maxRxnwl]     of float;
37  narray        = array[1..maxNumRxn]    of integer;
38  carray        = array[1..maxNumRxn]    of char;
39
40  Iarray        = array[1..NumSolWl]     of float;
41  ReactSet      = array[1..maxNumRxn]    of Rarray;
42  ReactIdStr    = string[80];
43  ReactIdSet    = array[1..maxNumRxn]    of ReactIdStr;
44
45  fluxrecord =
46   Record
47    Time  : float;
48    case b: boolean of
49      true:(lambda:Iarray);
50     false:(ival  :Iarray);
51   end;
52
53 VAR
54  i,j,l         : integer;
55 currWL,currRxn: integer;
56 srH,Time,
57 SpecTime      : float;
58
59 IRWL          : ReactSet;   {20 by 100 array of reaction wl}
60 ICSQY         : ReactSet;   {20 by 100 array of reaction csqy}
61 CQtype        : carray;     {type of spectrum P,B,C,E,b,c,e}
62 scalefactor   : float;      {some CsQy files have internal scalefactors}
```

```
63   numRxnWl      : narray;      (the number of wavelenghts for the reaction spectrum)
64   numRxn        : integer;     (the number of CsQy sets read in)
65   ReactId       : ReactIdSet; (id lines from reaction CsQy files)
66
67   IFWL          : iarray;      ( 1 by 93 array of irradiance wavelengths)
68
69   IFLUX         : fluxrecord; ( 1 by 93 array of integrated spectral flux
70                                   *all* at same IFWL wavelengths.)
71
72   SingleSpectrum, LoadedCQ     : boolean;
73
74   CmdLine : string;
75   Command : string[4];
76   OldWl   : float;
77   CQdefaultdir, IRRdefaultdir, LIXdefaultdir,
78   filename, switches : filenamestring;
79
80   FluxIn  : file of fluxrecord;
81   DataIn  : EText;
82   pfile   : PtrText;
83   DataOut : text;
84   obuf    : array[1..1024] of char;
```

08-24-88 21:42:36 CHMRATE.PAS
Mon 10-10-88 22:49:46 Sig_Error


```
 85  (<f>)
 86  Procedure Sig_Error(mess:string; showfile, action:boolean);
 87   var
 88    i: integer;
 89    line: string;
 90    oldattr: integer;
 91   begin
 92    OldAttr:=TextAttr;
 93    TextColor(Red);
 94    Writeln('ERROR:::',mess);
 95    if showfile
 96     then
 97      begin
 98       writeln('Next 3 lines were:');
 99       TextColor(Yellow);
100       for i:= 1 to 3 do
101        if not eof(pfile^)
102         then
103          begin
104           readln(pfile^,line);
105           writeln(line);
106          end;
107      end;
108    TextAttr:=OldAttr;
109    if action then halt(7);
110   end;
```

```
111  (<f>)
112  ($F+  make this one a FAR call)
113  Procedure ReadCsQy(Var FEOF: boolean; Var wl,csqy: float);
114  ($F-)
115   const
116    wF = 15; (width of Field)
117   var
118    StrF  : text;
119    Line  : string;
120    Dx    : string[wF]; (dummy var to read first wF-chars of line)
121    leof  : boolean;
122    cs, qy: float;
123    (global to this procedure are:
124      pfile--a pointer to text file to read spectrum
125      scalefactor--in-file adjustment factor for CS, normally 1.0
126    )
127   begin
128    (set up string 'Line' as a dummy file buffer)
129    AssignStr(StrF,Line);
130    (find a non-comment line and process FACtor lines)
131    repeat
132     leof:=eof(pfile^);
133     readln(pfile^, Line);
134     if (StrUp(copy(Line,1,3))='FAC')
135      then
136       begin
137        Line:=ParseFill(Line,' ','LR',2,wF);
138        Reset(StrF);
139        ($I-)read(StrF,Dx,scalefactor);($I+)
140        IOerrCode:=IOResult;
141        if IOerrCode<>0
142         then DisplayIOError(StrF,'read(StrF,Dx,scalefactor)','ReadCsQy');
143        Line:='';
144       end;
145    until ((Line[1]<>'!') and ((length(Line) > 0) or leof));
146    if not leof
147     then
148      begin
149       (parse the line and return three right-justified,wF-char fields)
150       Line:=ParseFill(Line,'x','RRR',3,wF);
151       i:=pos('x',Line);
152       (qy was not given in file, assume 1.0 )
153       if (i<>0) then Insert('1.0  ',Line,i-3);
154       (read three real numbers from the string Line)
155       Reset(StrF);
156       ($I-) read(StrF, wl, cs, qy); ($I+)
157       IOerrCode:=IOresult;
158       if (IOerrCode<>0)
159        then DisplayIOError(StrF,'read(StrF,wl,cs,qy)','ReadCsQy');
160       (Some files have an end of file signal ==> negative wl and cs)
161       if (wl < 0)
162        then
163         begin
164          FEOF:=true; (let calling procedure know we are finished)
165          wl:=0.0;
166          csqy:=0.0;
167         end
168        else
169         begin  (ok values read from file...)
170          if (wl < 4.0)  { assume microns instead of nm }
171           then wl := wl*1000;  { change to nm }
172          if wl<=oldwl
```

```
173           then Sig_Error('Wavelengths out of order!',true,stop)
174           else oldwl:=wl;
175          csqy := cs*qy*scalefactor;
176          FEOF := false;
177         end;
178       end (if not eof)
179      else
180       begin
181        FEOF:=true; (let calling procedure know we are finished)
182        wl:=0.0;
183        csqy:=0.0;
184       end;
185    end; (ReadCsQy)
```

```
186  {<f>}
187  procedure LoadReactSpec(NCsQy:integer; Var sfilename: filenamestring);
188   var
189    Line : string;
190    i,N  : integer;
191    FEOF : boolean;
192    {global to this procedure are:
193      pfile     -- file to read cs and qy from, assigned here
194      CQtype    -- the type of spectrum being read, assigned here
195      ReactID   -- the header in each reaction file, assigned here
196      numRxnWl  -- the number of wavelengths for reaction
197     calls:
198      IntegrateSpec  }
199   begin
200    ResetEText(DataIn,sfilename,EchoOFF);
201    pfile:=Cvt(DataIn);    {get the actual file pointer}
202    repeat
203     readln(pfile^,Line);
204    until (Line[1]<>'!');  {first non-comment is a Reaction Line}
205    writeln(Line);
206    ReactId[NCsQy]:='("'+Line+'")';
207    repeat
208     readln(pfile^,Line[1]);
209    until (Line[1] <> '!'); {second non-comment is the spectrum type}
210    CQtype[NCsQy]:=Line[1];
211    if NOT (CQtype[NCsQy] in ['P','B','C','E','p','b','c','e'])
212     then Sig_Error('React CQ File has incorrect PBCEpbce type.',showf,stop);
213    scalefactor:=1.0;
214    N:=maxRxnWl;
215    FEOF := false;
216    IntegrateSpec(@ReadCsQy,ModeSum,CQtype[NCsQY],N,IRWL[NCsQy],ICSQY[NCsQy]);
217    numRxnWl[NCsQy]:=N;
218    CloseEText(DataIn);
219   end;
```

```
220  (<f>)
221   Function BarBarRate(numIrrWl,numRxnWl: integer;
222                       Var IWx,IFx: iarray; Var RW,RP: Rarray):float;
223   { IFx is the value of the area in a bar of width IWx[j]-IWx[j-1],
224         eg, "ending interval" and average height IFx[j]/(IWx[j]-IWx[j-1]).
225     RP[i] is area for 0 to RW[i] of integrated reaction potential (original
226         spectra was 'bars', eg average values over interval.)
227     Function will calculate intergral averaged RP at IWx wavelengths,
228         ie, to match irradiance 'bars' and will then approximate
229         interval rate by the product of the averages. }
230    var
231     iwl,j,k : integer;
232     rate : float;
233     low_fwl, hi_fwl, wl_interval, incrate,
234     low_area_RP,
235     hi_area_RP,
236     IntFx, AvgRP : float;
237    begin
238     {find starting locations in irrad and csqy arrays}
239     j:=1; {index to irrad}
240     k:=1; {index to csqy}
241     while ((RW[k] < IWx[1]) and (k < numRxnWl)) do
242      k:=k+1;  {in case RW starts before IW}
243     if ((k > 1) and (k <= numRxnWl)) then dec(k);
244     while ((IWx[j] < RW[1]) and (j < numIrrWl)) do
245      j:=j+1;  {in case IW starts before RW}
246     if ((j > 1) and (j <= numIrrWl)) then dec(j);
247     if j=1
248      then low_fwl := IWx[j]-(IWx[2]-IWx[1])
249      else low_fwl := IWx[j-1];
250     low_area_RP := OneWay(RP,RW,numRxnWl,k,low_fwl,Search);
```

08-24-88 21:42:36 CHMRATE.PAS
Mon 10-10-88 22:49:46 BarBarRate


```
251  (<f>)
252     (In loop below, k keeps track of where we are in RW/RP while
253      j is incremented thru irrad spec wavelengths.)
254     rate:=0.0;
255     while ((k < numRxnWl) and (j < numIrrWl)) do
256      begin
257       hi_fwl       := IWx[j]; (get the hi end flux wavelength)
258       IntFx        := IFx[j]; (get the interval flux)
259       (interpolate RP area that corresponds to flux upper wavelength)
260       hi_area_RP   := OneWay(RP,RW,numRxnWl,k,hi_fwl,Search);
261       if (k < numRxnWl)
262        then
263         begin
264          wl_interval:=(hi_fwl - low_fwl);
265          AvgRP:= (hi_area_RP - low_area_RP)/wl_interval;
266          increate := (1E+16)*IntFx*AvgRP;
267          rate := increate + rate;
268          if SingleSpectrum
269           then
270            begin
271             writeln(DataOut,low_fwl:4:1,' ',
272                (IntFx):8:4,' ',(IntFx/wl_interval):8:4,' ',
273                (AvgRP*1E16):8:4,' ',increate:8:4,' ',rate:8:4);
274             writeln(DataOut,hi_fwl:4:1,' ',
275                (IntFx):8:4,' ',(IntFx/wl_interval):8:4,' ',
276                (AvgRP*1E16):8:4,' ',increate:8:4,' ',rate:8:4);
277            end;
278          (remember values for next time...)
279          low_fwl     := hi_fwl;
280          low_area_RP := hi_area_RP;
281         end;
282       inc(j);
283      end; (while)
284     BarBarRate:=rate; (return the accumulated product)
285    end;
```

```
286  (<f>)
287   Function BarLineRate(numIrrWl,numRxnWl: integer;
288                        Var IWx,IFx: Iarray; Var RW,RP: Rarray):float;
289   {IFx is the value of the area in a bar of width IWx[j]-IWx[j-1],
290         eg, "ending interval" and average height IFx[j]/(IWx[j]-IWx[j-1]).
291    (RP[k-1], RP[k]) are the points at the ends of a straight line segment.
292
293    The Operation is to apply Double Trapozoid Rule to common segments
294    formed by a union of wavelengths from both spectra.  Once the interval
295    wavelengths are determined, the IFx bar average height is computed and
296    becomes the value for both c and d in the double rule.
297   }
298    var
299     j,k : integer;
300     rate,
301     low_wl, hi_wl,
302     incrate, interval,
303     a,b,c : float;
304    begin
305     rate:=0.0;
306     {find starting locations in irrad and csqy arrays}
307     j:=1; {index to irrad}
308     k:=1; {index to csqy}
309     while ((RW[k] < IWx[1]) and (k < numRxnWl)) do
310      k:=k+1;  {in case RW starts before IW}
311     if ((k > 1) and (k <= numRxnWl)) then dec(k);
312     while ((IWx[j] < RW[1]) and (j < numIrrWl)) do
313      j:=j+1;  {in case IW starts before RW}
314     if ((j > 1) and (j <= numIrrWl)) then dec(j);
315     {begin by taking the first-1 Irradiance wavelength as left hand edge,
316      and interpolating the reaction potential spectrum for the beginning
317      interval wavelength (which is the previous interval ending wavelength).
318      Note that if j=1 then must calc a previous interval ending wavelength}
319     if j=1
320      then low_wl := IWx[j]-(IWx[2]-IWx[1])
321      else low_wl := IWx[j-1];
322     {In double trap rule, c=left-side  Irr spec bar height,
323                           d=right-side Irr spec bar height=c,
324                           a=left-side  RP  spec interpolated value,
325                           b=right-side RP  spec interpolated value.  }
```

```
326  (<f>)
327     (The code assumes that IFx *is* the area in the bar, therefore,
328      must calc the average height of the bar, given the width.       )
329     c := (1E+16)*IFx[j]/(IWx[j]-low_wl);
330     a := OneWay(RP,RW,numRxnWl,k,low_wl,Search);
331     while ((k < numRxnWl) and (j < numIrrWl)) do
332      begin
333       (find which spectrum contains next inflection point)
334       if (RW[k+1] < IWx[j])
335        then
336         begin
337          (the reaction spectrum has next inflection point)
338          inc(k);
339          hi_wl := RW[k];
340          interval := hi_wl-low_wl;
341          b := RP[k];
342         end
343        else
344         begin
345          (the irradiance spectrum has next inflection point)
346          hi_wl := IWx[j];
347          interval := hi_wl-low_wl;
348          (interpolate the reaction spectrum)
349          b := OneWay(RP,RW,numRxnWl,k,hi_wl,Search);
350         end;
351       incrate := (0.5*(a+b)*c)*(interval);
352       rate := incrate + rate;
353       if SingleSpectrum
354        then
355         begin
356          writeln(DataOut,low_wl:4:1,' ',
357             (c*interval*1E-16):8:4,' ',(c*1E-16):8:4,' ',
358             (0.5*(a+b) *1E+16):8:4,' ',incrate:8:4,' ',rate:8:4);
359          writeln(DataOut, hi_wl:4:1,' ',
360             (c*interval*1E-16):8:4,' ',(c*1E-16):8:4,' ',
361             (0.5*(a+b) *1E+16):8:4,' ',incrate:8:4,' ',rate:8:4);
362         end;
363       (remember values for next time...)
364       low_wl := hi_wl;
365       a := b;
366       if low_wl >= IWx[j]
367        then
368         begin
369          inc(j);
370          c := (1E+16)*IFx[j]/(IWx[j]-IWx[j-1]);
371         end;
372      end; (while)
373     BarLineRate:=rate; (return the accumulated product)
374    end;
```

```
375  (<f>)
376   Procedure InitVar;
377    var
378     i:integer;
379    begin
380     LoadedCQ:=false;
381     SingleSpectrum:=false;
382     for i:= 1 to maxNumRxn do numRxnWl[i]:=0;
383     numRxn:=0;
384     CQdefaultdir:='';
385     IRRdefaultdir:='';
386     LIXdefaultdir:='';
387    end;
388
389   Procedure CQCmd;
390    begin
391     filename:=copy(CmdLine,pos('=',CmdLine)+1,length(CmdLine));
392     if (length(CQdefaultdir) <> 0)
393      then filename:=CQdefaultdir+filename;
394     if NOT FileExists(filename)
395      then Sig_Error('Can not find CQ file '+filename,noshow,stop);
396     inc(numRxn);
397     OldWl:=0.0;
398     LoadReactSpec(numRxn,filename);
399     writeln('React CQ File Loaded...');
400     LoadedCQ:=true;
401     writeln('Number of reaction wavelengths :',numRxnWl[numRxn]:4);
402   end;
```

08-24-88 21:42:36 CHMRATE.PAS
Mon 10-10-88 22:49:46 CalcRates


```
403  {<f>}
404   Procedure CalcRates;
405    var
406     i,j,k : integer;
407     numtimes, limit : integer;
408     rate: float;
409    begin
410     for i:=1 to numRxn do
411      writeln(DataOut,ReactID[i]);
412     writeln(DataOut,'LIGHT=',numRxn:1,'#');
413     write(DataOut,'   0 (',CvtTimeSHHMM(srH),'): ');
414     for i:=1 to numRxn do
415      write(DataOut,'  0.0001, ');
416     writeln(DataOut);
417     numtimes := integer(FileSize(FluxIn));
418     if (numtimes<48) then limit:= numtimes
419                      else limit:= 48;
420     k:=1;
421     while (k<limit) do
422      begin
423       inc(k);
424       read(FluxIn,IFLUX);
425       with IFLUX do
426        begin
427         write(DataOut, ((Time-srH)*60.0):4:0,' (',CvtTimeSHHMM(Time),'): ');
428         For j:=1 to numRxn do
429          begin
430           if CQType[j]='P'
431            then
432             rate:=BarLineRate(NumSolWl,numRxnWl[j],IFWL,ival,IRWL[j],ICSQY[j])
433            else
434             rate:=BarBarRate (NumSolWl,numRxnWl[j],IFWL,ival,IRWL[j],ICSQY[j]);
435           if (rate<0.00009)
436            then write(DataOut,(0.0001):8:4)
437            else write(DataOut, rate:8:4);
438           if (j<numRxn)
439            then write(DataOut,', ')
440           else
441           if (k<limit)
442            then write(DataOut,', ')
443           else
444            write(DataOut,'; ');
445          end;
446        writeln(DataOut);
447       end; {with}
448      end; {while}
449     writeln(DataOut,'ENDX;');
450     Close(FluxIn);
451     Close(DataOut);
452    end;
```

08-24-88 21:42:36 CHMRATE.PAS Pg 13
Mon 10-10-88 22:49:46 SpectralRate of 16
453-482

```
453  (<f>)
454   Procedure SpectralRate;
455    var
456     rate: float;
457     j : integer;
458    begin
459     repeat
460      read(FluxIn,IFLUX);
461     until ( (((SpecTime-0.02(1min))<IFLUX.time) and
462                                    (IFLUX.time<(SpecTime+0.02(1min))) )
463             or eof(FluxIn) );
464     if eof(FluxIn)
465      then
466       Sig_Error('Found EOF(Irradiance) while looking for time '+CvtTimeSHHMM(SpecTime),false,stop);
467     for j:= 1 to numRxn do
468      begin
469       writeln(DataOut,ReactId[j]);
470       writeln(DataOut,'"T="',' ',CvtTimeSHHMM(IFLUX.time));
471       writeln(DataOut,'"B/E WL" ', '"IntrFX*E+16" ','"Fx/nm*E+16" ',
472                       '"AvgRP*E-16" ','"IncRate" ','"Rate" ');
473       if CQType[j]='P'
474        then
475         rate:=BarLineRate(NumSolWl,numRxnWl[j],IFWL,IFLUX.ival,IRWL[j],ICSQY[j])
476        else
477         rate:=BarBarRate (NumSolWl,numRxnWl[j],IFWL,IFLUX.ival,IRWL[j],ICSQY[j]);
478       writeln(DataOut);
479      end;
480     close(FluxIn);
481     close(DataOut);
482    end;
```

```
483  {<f>}
484   Procedure IrrCmd;
485    var
486     i,L: integer;
487     filename,fluxfilename,lixfilename:filenamestring;
488    begin
489     if NOT LoadedCQ
490      then Sig_Error('No CS QY reaction data loaded yet.',noshow,stop);
491     filename:=copy(CmdLine,pos('=',CmdLine)+1,length(CmdLine));
492     fluxfilename:=IRRdefaultdir+filename;
493     Assign(FluxIn,fluxfilename);
494     {$I-}Reset(FluxIn);{$I+}
495     if IOResult<>0
496      then Sig_Error('Can not find Irradiance File '+fluxfilename,noshow,stop);
497     writeln('Chamber Spectral Irradiance file found...');
498     if (LIXdefaultdir='')
499      then lixfilename:=IRRdefaultdir+filename
500      else lixfilename:=LIXdefaultdir+filename;
501     L:=pos('.',lixfilename);
502     if SingleSpectrum
503      then
504       begin
505        lixfilename[L+1]:='S';
506        lixfilename[L+3]:='R';
507       end
508      else
509       begin
510        lixfilename[L+1]:='L';
511        lixfilename[L+3]:='X';
512       end;
513     OpenFile(DataOut,'Creating file',lixfilename,switches,Obuf,SizeOf(Obuf),'W');
514     read(FluxIn,IFLUX);
515     srH:=IFLUX.time; {sunrise is first time in file}
516     for i:=1 to NumSolWl do IFWL[i]:=IFLUX.ival[i];
517     if SingleSpectrum
518      then SpectralRate
519      else CalcRates;
520    end;
```

```
521  {<f>}
522   Procedure SingCmd;
523    begin
524     SingleSpectrum:=true;
525     SpecTime:=CvtSHHMMTime(copy(CmdLine,pos('=',CmdLine)+1,4));
526     readln(Ext(DataIn)^,CmdLine);
527     Command:=StrUp(copy(CmdLine,1,4));
528     writeln(CmdLine);
529     IrrCmd;
530     SingleSpectrum:=false;
531    end;
532
533   Procedure ShowData;
534    var
535     filename,switches: filenamestring;
536     i,j : integer;
537    begin
538     filename:=copy(CmdLine,pos('=',CmdLine)+1,length(CmdLine));
539     OpenFile(DataOut,'Creating File',filename,switches,Obuf,SizeOf(Obuf),'W');
540     writeln(DataOut,'"The Integrated CS and QY"');
541     for i:= 1 to numRxn do
542      begin
543       writeln(DataOut,'"',ReactId[i],'"');
544       for j:= 1 to numRxnWl[i] do
545        writeln(DataOut,j:3,' ',IRWL[i,j]:6:2,' ',(ICSQY[i,j]*1E16):8:4);
546       writeln(DataOut);
547      end;
548     Close(DataOut);
549    end;
```

```
550  (<f>)
551  begin  ( main )
552   TextBackground(Black);
553   ClrScr;
554   TextColor(Yellow);
555   writeln(title);
556   TextColor(Red);
557   writeln(copyright);
558   TextColor(Green);
559   writeln(version);
560   writeln;
561   writeln('Expected Command Line Input:');
562   writeln(' ChmRate  cmd_file_name  ');
563   writeln;
564   TextColor(LightCyan);
565
566   (Extended Text pointer should be initialized before first call)
567   DataIn := InitEText;
568   if ParamCount >= 1
569    then filename := ParamStr(1)
570    else filename := '';
571   OpenEText(DataIn, 'Enter name of command file ',filename,switches);
572
573   Window(1,5,79,25);
574   TextColor(White);
575   ClrScr;
576   InitVar;
577
578   repeat
579    readln(Ext(DataIn)^,CmdLine);
580    Command:=StrUp(copy(CmdLine,1,4));
581    writeln(CmdLine);
582    if (Command='CSQY') then CQCmd
583    else
584    if (Command='IRRA') then IrrCmd
585    else
586    if (Command='SING') then SingCmd
587    else
588    if (Command='CQDE')
589     then CQdefaultdir:=copy(CmdLine,pos('=',CmdLine)+1,length(CmdLine))+'\'
590    else
591    if (Command='IRRD')
592     then IRRdefaultdir:=copy(CmdLine,pos('=',CmdLine)+1,length(CmdLine))+'\'
593    else
594    if (Command='LIXD')
595     then LIXdefaultdir:=copy(CmdLine,pos('=',CmdLine)+1,length(CmdLine))+'\'
596    else
597    if (Command='DATA')
598     then ShowData
599    else
600    if ((length(CmdLine)=0) or (Command='EXIT'))
601     then ( blank line --- just skip it )
602    else
603     writeln('Unknown Command.');
604   until (Command='EXIT');
605
606   CloseEText(DataIn);
607   writeln;
608   writeln('Finished.');
609  end.
```

```
Mon 10-10-88 22:49:46  INDEX (Cross Ref) Page.Line
 procs, fcns, external idents
  1 08-24-88 21:42:36  CHMRATE.PAS

BarBarRate          <7.221    8=284    12.434   13.477
BarLineRate         <9.287   10=373    12.432   13.475
Black       16p552
b            1.48    9.303   10=341   10=349   10.351   10.358   10.361   10.365
CalcRates           <12.404   14.519
ChamRate     1.1
CloseEText             16.606
ClrScr      16.553     16.575
CmdLine      2.74    11p391   14p491   15p525   15p526   15p527   15p528   15p538   16p579
            16p580   16p581   16p589   16p592   16p595   16p600
Command      2.75    15=527   16=580   16.582   16.584   16.586   16.588   16.591   16.594
            16.597   16.600   16.604
CQCmd      <11.389   16.582
CQdefaultdir           2.77    11=384   11p392   11.393   16=589
CQtype       1.61     6.210    6.211    6.216
Crt          1.3
carray       1.38     1.61
copyright              1.19   16p557
currRxn      1.55
currWL       1.55
DataIn       2.81     6p200    6p201    6p218   15p526   16=567   16p571   16p579   16p606
DataOut      2.83     8p271    8p274   10p356   10p359   12p411   12p412   12p413   12p415
            12p416   12p427   12p436   12p437   12p439   12p442   12p444   12p446   12p449
            12p451   13p469   13p470   13p471   13p478   13p481   14p513   15p539   15p540
            15p543   15p545   15p546   15p548
Dos          1.4
ErrIO        1.15
EscChk       1.14
EText        2.81
FileOpen     1.5
FluxIn       2.80    12p417   12p424   12p450   13p460   13p463   13p464   13p480   14p493
            14p494   14p514
filename     2.78    11=391   11=393   11p394   11.395   11p398   14.487   14=491   14.492
            14.499   14.500   15.535   15=538   15p539   16=569   16=570   16p571
filenamestring         2.78     6.187   14.487   15.535
float        1.34     1.36     1.40     1.47     1.57     1.62     2.76     4.113    4.122
             7.222    7.232    7.236    9.288    9.303   12.408   13.456
fluxrecord             1.45     2.69     2.80
Green       16p558
HuntFind     1.8
HuntTab      1.7
Iarray       1.40     1.49     1.50     9.288
ICSQY        1.60     6.216   12.432   12.434   13.475   13.477   15.545
IFLUX        2.69    12p424   12.425   13p460   14p514
IFWL         2.67    12p432   12p434   13p475   13p477   14.516
InitEText              16.567
InitVar    <11.376     16.576
IRRdefaultdir          2.77    11=385   14.492   14.499   16=592
IRWL         1.59     6.216   12.432   12.434   13.475   13.477   15.545
IrrCmd     <14.484    15.529   16.584
i            1.54     3.88     3=100    4=151    4.153    6.190   11.378   11=382   12.406
            12=410   12.411   12=414   14.486   14=516   15.536   15=541   15.543   15.544
            15.545
iarray       2.67     7.222
ival         1.50    12p432   12p434
j            1.54     7.231    7=239    7.244    7=245    7p246    7.247    7.248    7.249
             8.255    8.257    8.258    8p282    9.299    9=307    9.312    9=313    9p314
             9.319    9.320    9.321   10.329   10.331   10.334   10.346   10.366   10p369
            10.370   12.406   12=428   12.430   12.432   12.434   12.438   13.457   13=467
            13.469   13.473   13.475   13.477   15.536   15=544   15.545
```

Mon 10-10-88 22:49:46 INDEX (Cross Ref) Page.Line
procs, fcns, external idents

LIXdefaultdir 2.77 11=386 14.498 14.500 16=595
LightCyan 16p564
LoadedCQ 2.72 11=380 11=400 14.489
LoadReactSpec <6.187 11.398
l 1.54
lambda 1.49
MathFunc 1.13
ModeInc 1.31
ModeSum 1.30 6p216
maxNumRxn 1.22 11.382
maxRxnWl 1.23
NumSolWl 1.24 12p432 12p434 13p475 13p477 14.516
narray 1.37 2.63
noshow 1.28 11p395 14p490 14p496
numRxn 2.64 11=383 11p396 11p398 11.401 12.410 12.412 12.414 12.428
12.438 13.467 15.541
numRxnWl 2.63 6.217 7.221 7.241 7.243 7p250 8.255 8p260 8.261
9.287 9.309 9.311 10p330 10.331 10p349 11.382 11.401 12.432
12.434 13.475 13.477 15.544
OldWl 2.76 11=397
OpenEText 16.571
obuf 2.84
ParamCount 16.568
ParamStr 16.569
PtrText 2.82
pfile 2.82 3.101 3.104 4.132 4.133 6=201 6.203 6.208
Rarray 1.36 1.41 7.222 9.288
ReactId 2.65 6.206 13.469 15.543
ReactIdSet 1.43 2.65
ReactIdStr 1.42 1.43
ReactSet 1.41 1.59 1.60
ReadCsQy <4.113 6.216
ReadEtxt 1.6
Red 16p556
ShowData <15.533 16.598
Sig_Error <3.86 5.173 6.212 11.395 13.466 14.490 14.496
SingCmd <15.522 16.586
SingleSpectrum 2.72 8.268 10.353 11=381 14.502 14.517 15=524 15=530
SpecLib 1.9
SpecTime 1.57 13.461 13.462 13p466 15=525
SpectralRate <13.454 14.518
StrIO 1.11
Strings 1.10
scalefactor 1.62 4p139 5.175 6=213
showf 1.27 6p212
single 1.34
srH 1.56 12p413 12.427 14=515
stop 1.26 5p173 6p212 11p395 13p466 14p490 14p496
switches 2.78 14p513 15.535 15p539 16p571
TextBackground 16.552
TextColor 16.554 16.556 16.558 16.564 16.574
Time 1.47 1.56 12p427
TimeIO 1.12
title 1.18 16p555
version 1.20 16p559
White 16p574
Window 16.573
writeln 16.555 16.557 16.559 16.560 16.561 16.562 16.563 16.581 16.603
16.607 16.608